الفصل الثالث
أدوات ولغات البرمجة

يمتلك المبرمجون مجموعة من أدوات البرمجة ، ويقل عدد المبرمجين الذين يستخدمون لغة واحدة أو أداة برمجة واحدة ، إذ يحاول العديد منهم الوصول إلى قدر من الاحتراف بامتلاك مهارة استخدام أكثر من أداة .

لما كان الكمبيوتر لا يعرف سوى لغة الآلة فإنه يجب تحويل الأوامر المكتوبة بلغة من لغات البرمجة إلى لغة الآلة ، يتم هذا التحويل وفق أسلوبين :

- الأول منهما هو أسلوب المفسر Interpreter .
- والثانى منهما هو أسلوب المترجم Compiler .

وقد تحتوى لغة البرمجة الواحدة على مفسر ومترجم فى آن واحد .

الفرق بين المفسر والمترجم هو : أن المفسر يقوم بتفسير الأوامر (أو الجمل البرمجية) واحدا بعد الآخر ، إن هذا يشبه وجود كتاب باللغة الفرنسية ، ويقوم مترجم بقراءة السطور أو الفقرات ، وتحويلها إلى اللغة العربية لأشخاص لا يعرفون سوى اللغة العربية .

على الناحية الأخرى فإن المترجم فى لغات البرمجة يقوم باستلام البرنامج كله وتحويله إلى نص كامل بلغة الآلة ، إن هذا يشبه ترجمة الكتاب الفرنسى كله مرة واحدة إلى اللغة العربية ليستطيع الشخص العربى قراءة الكتاب بلغة عربية مرة واحدة دون ضرورة انتظار التفسير سطرا بعد سطر .

يبدو من هذا أن المترجم أكثر سرعة فى التنفيذ من المفسر ، وهذا هو الواقع .

إن المفسر يبين النتائج بشكل سريع ، ويمكن معه تعديل البرنامج لأنه لا يزال مكتوبا بلغة البرمجة ، أما المترجم فيحول البرنامج كله مرة واحدة إلى برنامج مكتوب بلغة الآلة .

عندما يعمل البرنامج عن طريق المفسر فإنه فى كل مرة يجب تفسير سطور البرنامج ، ويجب تواجد المفسر شغالا فى الكمبيوتر ليتمكن البرنامج من تفسير الأوامر .

عـــندما يعمـــل البرنامج بعد ترجمته فليست هناك حاجة لإعادة الترجمة ، فقد تحول الـــبرنامج إلى لغـــة الآلة التى يفهمها الكمبيوتر ، كما أنه ليست هناك حاجة لتشغيل بـــرنامج الـــترجمة فى جهـــاز الكمبيوتر أثناء تشغيل البرنامج ، فعند اكتمال عملية الـــترجمة يقـــوم المـــترجم بضم كل الشفرات اللازمة لتشغيل المترجم منفردا بذاته ، وربط عناصره مع الأدوات الأخرى ، وبالتالى يصبح برنامجا مستقلا .

نماذج لغات البرمجة

كيف بدأت لغات البرمجة ، وكيف تطورت ، وكيف تعمل لغة البرمجة ، وكيف يتم تخزين البرامج والبيانات فى الكمبيوتر على مستوى الآلة ؟ وكيف تخزن البرامج والبيانات ، وما هو الرقم الثنائى أو البت BIT ، والحرف أو البايت Byte ، وجدول رموز آسكى ASCII ، وما هى أنواع البرامج المناسبة للغات البرمجة؟ .

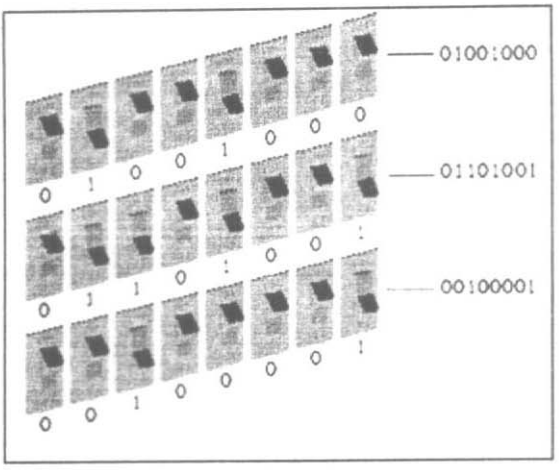
تخزين البرامج والبيانات

عند كتابة برنامج فى المحرر تضغط على المفاتيح ، فتذهب الحروف إلى أماكن فى ذاكرة الكمبيوتر ، ويخزن الكمبيوتر تمثيلا للحروف ، إن ذاكرة الجهاز مكونة من وحدات تخزين ، فعند شراء كمبيوتر بذاكرة سعتها ٣٢ ميجا بايت MB فذلك يعنى أن الكمبيوتر يمكنه تخزين ٣٢ مليون حرف تقريبا .

إن الكمبيوتر فى مستواه الأدنى هو ملايين من المفاتيح التى تقوم بتشغيل الكهرباء أو إيقافها ، وكل حرف يتم تمثيله فى الكمبيوتر بمجموعة من المفاتيح فى إحدى حالتين : تعمل On أو لا تعمل Off .

يعبر دائما عن المفتاح الذى يعمل برقم واحد 1 ، والذى لا يعمل برقم صفر 0 ، ونظرا لأن تلك المفاتيح لها إحدى قيمتين فقط إما 0 أو 1 ، لذلك تسمى هذه الأصفار والآحاد أرقاما ثنائية Binary Digits ، أو ما يعرف باسم بت BIT للاختصار ، ولكل حرف فى الكمبيوتر هناك ٨ بت (بايت Byte) ، ولهذا فإن كل حرف يأخذ ٨ بت لتمثيله أى ثمانية مفاتيح تكون Off أو On .

عند كتابة جميع التركيبات المحتملة لثمانى وحدات من الأصفار والآحاد بدءا من التركيب 00000000 حتى التركيب 11111111 .



تم إسناد حرف واحد إلى كل تركيب أصبح الجدول قياسيا يعرف باسم كود أو شفرة الآسكى ASCII ، اختصار عبارة (الشفرة الأمريكية القياسية لتبادل المعلومات American Standard Code For Information Interchange) مثال :

الرمز Character	شفرة آسكى	العدد العشرى المكافئ
Space	00100000	32
0	00110000	48
1	00110001	49
2	00110010	50
3	00110011	51
9	00111001	57
?	00111111	63
A	01000001	65
B	01000010	66
C	01000011	67
A	01100001	97
B	01100010	98

لكل قيمة من جدول رموز آسكى ASCII هناك رقم عشرى مقابل يرتبط بها ، ولهذا بالرغم من أن الكمبيوتر يمثل علامة الحرف A برموز 01000001 فيمكن الرجوع إلى تلك القيمة أثناء البرمجة بالرقم 65 ، وسوف يعرف الكمبيوتر مقصد المبرمج بمجموعة الرموز 010000001 أو الحرف A ، وإحدى مميزات لغات البرمجة عالية المستوى هى أنها تتيح استخدام النظام العشرى للأرقام ، وتقوم لغة البرمجة بتحويل القيمة العشرية إلى المكافئ الثنائى (٨بت) المستخدم داخل الكمبيوتر .

فى قيم آسكى ASCII لكل حرف فى الكمبيوتر قيمة آسكى ASCII منفردة خاصة به ، وتلك هى الطريقة التى يستطيع بها الكمبيوتر أن يميز بين الحروف المختلفة .

يستخدم الكمبيوتر الشخصى PC والكمبيوتر الصغير Mini جدول رموز ASCII أما الكمبيوتر الكبير Main Frame فيستخدم جدولا مشابها يسمى إبسديك EBCDIC ، ويعتبر جدول الآسكى ASCII هو طريقة التمثيل الأساسية لجميع البيانات والبرامج التى يتعامل معها الكمبيوتر .

عند ضغط حرف A لن يتم تخزين حرف A نفسه فى الكمبيوتر ، وبدلا منه فإن قيمة شفرة الآسكى ASCII المقابلة لحرف A هى التى تخزن ، ويتم تمثيل الحرف A بالقيمة 01000001 ، فعند ضغط حرف A فى لوحة المفاتيح تذهب قيمة ASCII وهى 01000001 إلى الذاكرة ، ويحفظ الكمبيوتر هذا الخليط من الأصفار والآحاد على هيئة مفاتيح On و Off فى ذلك الموقع من الذاكرة طالما بقى حرف A هناك .

1 byte

1	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0
1	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	1
1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1
0	0	0	0	0	1	1	1

عند طباعة الحرف على الطابعة يقوم معالج الكمبيوتر CPU بإرسال شفرة آسكى ASCII للحرف A إلى الطابعة التى تبحث عن القيمة 01000001 فى جدول رموز ASCII خاص بها ثم تطبع الحرف على الورق ، وبدءا من الوقت الذى غادر فيه حرف A لوحة المفاتيح إلى ما قبل طباعته لم يكن الحرف A موجودا على الإطلاق ، لكن كانت هناك مجرد تركيبة من الآحاد والأصفار التى تمثل حرف A .

الحساب الثنائى

لا يستطيع الكمبيوتر (فى مستواه الداخلى الأدنى) إجراء عمليات الطرح أو الضرب أو القسمة ، فأجهزة الكمبيوتر تستطيع فقط أن تقوم بالجمع ، فهى تجرى عملية الجمع على مستوى البت .

يعتبر الحساب الثنائى Binary هو الطريقة الوحيدة لحسابات الكمبيوتر ، ويمكن أن يقوم الكمبيوتر بالجمع فقط ، ويجعلنا نتصور أنه يجرى جميع أنواع العمليات الحسابية لأنه يفعل ذلك بسرعة هائلة ، وعندما تطلب من الكمبيوتر (عبر البرمجة) جمع أرقام يعود بالنتيجة فورا ، ولا توجد مشكلة فى الجمع .

تأتى المشكلة عند إجراء نوع آخر من العمليات الحسابية مثل الطرح ، لما كان الكمبيوتر يستطيع الجمع فقط ، ولا يمكنه الطرح ، يعمل الكمبيوتر شيئا آخر ، إذ يقوم بتمثيل الأرقام السالبة بأخذ القيمة السالبة للرقم ، ويقوم بتمثيلها على مستوى البت ، وبعد ذلك يمكنه "جمع" الرقم السالب إلى الرقم الثانى .

يقوم الكمبيوتر بتحويل الرقم إلى قيمته السالبة عبر عملية معروفة باسم المكمل الثنائى 2's Complement ، فالمكمل الثنائى لرقم هو القيمة السالبة له على مستوى البت .

يمتلك الكمبيوتر الدوائر الداخلية المنطقية القادرة على تحويل الرقم سريعا إلى المكمل الثنائى ، ثم يبدأ جمع الأرقام السالبة ، وبذلك يجرى عملية الطرح ظاهريا .

بعدما يقوم الكمبيوتر بإجراء عملية الجمع ومحاكاة عملية الطرح (من خلال عمليات متتالية من جمع الأرقام السالبة) ، يمكن أيضا أن يحاكى عملية الضرب والقسمة فلضرب رقمين يقوم الكمبيوتر فعليا بجمع الرقم الأول بعدد مرات الرقم الثانى لينتج الرقم .

فى القسمة يقوم الكمبيوتر بطرح العدد الثانى من العدد الأول (أى يقوم بجمع سالب العدد الثانى على العدد الأول) إلى أن يصل إلى رقم صفر ، ويبدأ فى إحصاء عدد المرات التى استغرقها فتكون الناتج .

يقوم الكمبيوتر بتمثيل الأرقام بطريقة تشبه تمثيل الحروف ، وبعدما تصل الأرقام إلى حد معين (٢٥٦ تحديدا) يستخدم الكمبيوتر أكثر من بايت واحد تمثيل الرقم شاغلا أى

عدد يحتاجه من مواقع الذاكرة .

البرامج الأولى

فى بداية أجهزة الكمبيوتر لم تكن برمجتها تتم بنفس طريقة كمبيوتر اليوم ، بل كان الجهد المبذول فيها كبيرا للغاية إذ كان الكمبيوتر القديم يحتفظ فقط بالبيانات وليس بالبرامج ، وكان مفهوم البرمجة وقتها مختلفا حيث كان يتم إدخال البرامج إلى قلب الآلة مباشرة Hard Wired ، فيتم إدخال الكود فعليا باستخدام التوصيلات بواسطة متخصصين وذلك لمعالجة البيانات ، إذ لم يكن يعرف المبرمجون الأوائل لوحة المفاتيح أو المحرر Editor أو المترجم Compiler ، وبذلك كانوا خبراء فى العتاد Hardware ، ولم يكونوا متخصصى برامج Software .

لعمل تغيير فى تلك البرامج كان على مبرمج العتاد أن يعيد توجيه مسار التوصيلات لأداء مهمته ، ولم يمض وقت طويل حتى اخترع فون نيومان مبدأ البرنامج المشارك Shared Program حيث أوضح أن البرنامج يمكن تخزينه فى الذاكرة بجانب البيانات ، وبعدها أصبحت البرامج أسهل فى التعديل .

كانت برمجة الكمبيوتر تستلزم جهدا فائقا حيث يجب أن تبرمج بلغة الآلة (آحاد وأصفار) ، وهكذا فقد تبدو الأوامر الأولى فى برنامج مثل هذا :

01000110
11000100
10111011
00011101
.
.

كلما أراد المبرمج جمع رقمين أو نقل القيم إلى الذاكرة كان عليه الرجوع إلى جدول يصف التركيبات الصحيحة من الآحاد والأصفار للأوامر المطلوبة ، مما كان يستغرق ساعات طويلة ، لكنها كانت خطوة هائلة للأمام انتقالا من البرمجة بطريقة التوصيلات المباشرة .

تبسيط لغة الآلة Machine Language

بدلا من كتابة الأصفار والآحاد تمكن المبرمجون من كتابة أسماء مرتبطة بكل أمر

فى لغة الآلة ، وهكذا فالأوامر السابقة أصبحت تبدو كالآتى :

```
ADD A, 17
MOV A, OUT
LOAD B
SUB B, A
```

صحيح أن مثل هذه الأوامر لا زالت مشفرة لكنها أسهل فى التذكر من الآحاد والأصفار ، وتسمى الكلمات المختصرة Mnemonics التى تعبر عن الأوامر ، فمثلا أمر ADD A , 17 سيكون أبسط كثيرا عند جمع رقم ١٧ على موقع الذاكرة A بدلا من كتابة رموز الأصفار والآحاد .

لا يمكن للكمبيوتر فهم هذه الأوامر المختصرة لكن يوجد برنامج مترجم يسمى التجميع Assembler يقوم بمهمة ترجمة الأوامر إلى الآحاد والأصفار التى يفهمها الكمبيوتر ويستطيع تمثيلها ، وعندما أصبحت لغة التجميع Assembler هى وسيلة إدخال الأوامر للكمبيوتر كلغة ثانية (بعد لغة الآحاد والأصفار) أتاحت سرعة أكبر فى تطوير البرامج .

البرمجة على المستوى الأدنى (لغات المستوى المنخفض)

تندرج هاتان اللغتان : لغة الآلة Machine Code ، ولغة التجميع Assembler تحت فئة لغات البرمجة على المستوى الأدنى (لغات المستوى المنخفض) Low Level ، إذ لا يحتاج الكمبيوتر لترجمة الكثير من أجل تحويل لغة التجميع إلى لغة الآلة ، حيث إن كل أمر مختصر يمتلك ترجمة مباشرة إلى لغة الآلة المكافئة .

فى الأغلب تحتوى الأجهزة المنزلية على كمبيوتر صغير بداخلها ، وتحتاج تلك الأجهزة إلى البرمجة المدمجة Embedded Programming ، ولخفض سعرها وتصغير حجمها يلجأ منتجوها إلى استخدام لغة الآلة فى برمجتها .

يستخدم مبرمجو لغة الآلة أسلوب تشفير آخر يعتمد على النظام السداسى العشرى Hexadecimal (القائم على الرقم 16) بدلا من النظام الثنائى (المبنى على الرقم 2) ويكون أسهل نسبيا فى التعامل والبرمجة .

هناك القليل من برامج ترجمة لغة التجميع Assembler الموجودة الآن ومعظمها تكون متاحة فقط بشرائها حيث يقل الطلب عليها ، وأكثرها شيوعا هو برنامج

Macro Assembler من مايكروسوفت على الرغم من إمكانية التعامل مع برنامج التجميع Assembler من لغات البرمجة المرئية ذات المستوى الأعلى مثل Visual C ، وتتطلب عناصر لغة التجميع الكثير من التعقيد والصعوبة .

مع البيئة الموجودة اليوم من اتصال الكمبيوتر بالشبكات أو بالإنترنت أصبح الأمن مهما عند التعامل مع كود فى مستوى لغة الآلة فنظم التشغيل مثل ويندوز تعمل جميعها على حماية الجهاز من محاولات تدمير نظام الأمن وتعديلات الذاكرة المقصودة أو المتعمدة ، لذا تقوم نظم التشغيل على منع تغيير محتويات منطقة النظام فى الذاكرة بدون حيازة صلاحيات كافية على النظام ، وعلى ذلك فقد لا يمكن التعامل مع مناطق معينة فى الذاكرة عند البرمجة بلغة الآلة أو بلغة التجميع .

البرمجة على المستوى العالى (لغات المستوى العالى)

تطورت لغات البرمجة العالية المستوى فبعد وجود لغة التجميع Assembler بدأت شركات أكثر فى استخدام الكمبيوتر ، ومع انتشار الكمبيوتر دعت الحاجة إلى كتابة تطبيقات وبرامج أكثر تعقيدا لهذه الشركات ، ولم تكن لغة التجميع Assembler فعالة بما يكفى لهذا التطور السريع فبدأ تطوير لغات عالية المستوى High Level عملت على إضافة طبقة وسيطة أخرى بين المبرمج والجهاز .

كانت تلك الطبقة الإضافية تعنى المزيد من العمل المطلوب من الكمبيوتر لترجمة لغة البرمجة عالية المستوى High Level إلى لغة الآلة ، لكن أصبح المبرمج فى حال أفضل إذ تحرر من عبء الكتابة بلغة الآلة ، وتعتبر كل من لغة فورتران FORTRAN ، ولغة كوبول COBOL من أقدم لغات البرمجة عالية المستوى ، وما زالتا تستخدمان على حتى الآن .

لغة فورتران FORTRAN

كلمة فورتران Fortran هى اختصار مترجم المعادلة Formula Translator ، فهى لغة برمجة تستخدم فى التطبيقات الرياضية والعلمية ، وتعمل لغة فورتران FORTRAN بصورة جيدة مع الأرقام عالية الدقة ، وتوفر مكتبة ممتازة من الدوال المدمجة لحساب المثلثات التى تساعد فى الحقل العلمى .

عبر سنوات أضاف المبرمجون إلى لغة فورتران FORTRAN لإعطائها المزيد من قدرات التعامل مع الحروف ، فقد كانت الإصدارات الأولى منها تقوم بحل العمليات الرياضية دون الاكتراث بمظهر النتائج ، ومازالت اللغة تعمل حتى اليوم للكمبيوتر الكبير Main Frame والكمبيوتر الشخصى PC .

عند استعراض نموذج برنامج لغة فورتران FORTRAN سوف تكتشف التشابه بينها وبين لغات البرمجة عالية المستوى الأخرى .

بالرغم من أن لغة فورتران FORTRAN هى لغة عالية المستوى أسهل فى القراءة من لغة التجميع لكنها قد تبدو مبهمة ، لكن يمكن فهم كلمات مألوفة مثل (اكتب) WRITE ، وتنسيق FORMAT .

هكذا أصبحت اللغات عالية المستوى للبرمجة أقرب للغة الإنجليزية ، وتحتوى لغة فورتران على عدد قليل نسبيا من الأوامر (بالمقارنة بلغة كوبول COBOL ولغة بيزك Basic) ، كما لا تعتبر لغة فورتران من اللغات التى تفسر نفسها أو تساعد غير المبرمجين على فهمها ، ورغم ذلك فقد أثبتت جدارة فى المجال العلمى لكنها فقدت بعض مكانتها أمام لغات مثل باسكال Pascal ولغة سى C .

لغة كوبول COBOL

وضعت لغة كوبول COBOL فى عام ١٩٦٠ بسبب الحاجة إلى لغة أخرى لمجال الأعمال بدلا من لغة فورتران التى كانت تقوم بالحسابات العلمية لكنها لم تصمم للتعامل مع مجال الأعمال ، كما اكتشف المبرمجون أن طبيعة شفرة لغة فورتران كانت تبطئ من مهام صيانة البرامج .

لغة كوبول COBOL اختصار كلمات اللغة الموجهة للأعمال العامة Common Business Oriented Language ، وكان هدفها الأساسى هو لغة سهلة الفهم تعالج كمية كبيرة من بيانات مجال الأعمال مثل سجلات العاملين أو المخازن .

تنقسم جميع برامج COBOL إلى أربعة أقسام Divisions : يبدأ قسم التعريف Identification بوصف البرنامج ، أما قسم البيئة Environment فهو يصف نظام الكمبيوتر الذى يعمل عليه البرنامج ، ويصف قسم البيانات Data هيئة جميع البيانات المستخدمة وتنسيقها فى البرنامج ، وأخيرا يحتوى قسم الإجراءات Procedure على

الكود اللازم لمعالجة البيانات ، وتعتبر لغة كوبول أفضل من لغة فورتران فى تطبيقات مجال الأعمال ، ويكون البرنامج تقريبا ضعف طول برنامج فورتران .

بدلا من أن تكون لغة كوبول COBOL سهلة الفهم وتشرح نفسها انتهت إلى أن أصبحت لغة كثيرة المفردات إلى حد إعاقة رؤية الكود الأصلى ، ويتفق الكثيرون على أنها لا تحقق مبدأ الشرح الذاتى ، ورغم ذلك فقد احتلت المركز الأول فى لغات البرمجة لمجال الأعمال خلال السنوات الماضية ، ومازالت شائعة حتى الآن بالرغم من أن البرامج الجديدة بلغات أخرى تحل محل تلك الأقدم المكتوبة بلغة كوبول COBOL ، لكن بمعدل بطئ فمعظم الهيئات الكبيرة وبعض الشركات مازالت تستخدم كوبول لمعالجة البيانات .

إذا أراد مبرمج أن يعمل بالبرمجة كوظيفة عليه التخطيط لتعلم لغة كوبول COBOL كلغة ثانية أو ثالثة باستخدام أحد إصدارات الكمبيوتر الشخصى المتاحة لفهم البرامج المستخدمة حاليا ، ويزداد الطلب عليه عندما تقرر الشركات استبدال الكود القديم بلغة كوبول COBOL بآخر أحدث .

لغات أخرى

بعد أن سيطرت لغة فورتران FORTRAN ثم لغة كوبول COBOL على الساحة بدأت تظهر لغات أخرى بكثافة ، حيث أمكن للشركات تطوير لغة برمجة خاصة بها تدعم محيط العمل بالشركة بصورة أفضل من اللغتين الكبيرتين .

لغة PL/1

أخذت لغات عديدة فى الظهور إلى الحد الذى أصبح معه مجتمع البرمجة فى حيرة وغير قادر على اتخاذ قرار بشأن لغة البرمجة الأنسب للاستخدام فى أى مشروع ، حتى قررت شركة آى بى ام IBM تصميم "لغة البرمجة الوحيدة التى يحتاجها أى شخص" تقدم كل شئ ، وأفضل لغة علمية ، وأفضل لغة للأعمال ، وتحل جميع المشكلات ، وتختص بمتطلبات أى مبرمج ، فقدمت لغة البرمجة الأولى PL/1 اختصار Programming Language 1 لتحل مشكلة اللغات الكثيرة .

قامت الشركة بتصميمها آخذه أفضل ما فى كوبول وأفضل ما فى فورتران ومجموعة

فى اللغات الأخرى فى ذلك الوقت ، ولم تتحقق النتيجة النهائية على الأقل من وجهة نظر المبيعات ، ولم يكتب النجاح الذى كانت تأمله الشركة مع لغة البرمجة PL/1 ، وأصبحت مجرد لغة أخرى ضمن العديد منها .

المشكلة الأساسية فى لغة PL/1 أنها كانت أفضل من اللازم للاستفادة بجميع لغات البرمجة المعروفة وقتها ، فقد كانت لغة برمجة ضخمة للغاية ، تستلزم موارد هائلة لتشغيلها وخلال الستينات لم يكن نطاق الأعمال يملك قدرة كمبيوتر كافية لتخصيص كل وقت المعالج لترجمة برنامج لغة PL/1 ، كذلك كانت تأخذ وقتا طويلا من المبرمجين لتعلمها إذ أنها كانت ضخمة بحيث كان من النادر أن تجد من هو متمكن منها كما يجب .

سوف تجد بعض اليوم الشركات لديها برامج PL/1 والقليل مازالت تكتب برامج باستخدامها لكنها لم تلق الاستحسان المأمول .

تشبه لغة PL/1 لغة كوبول COBOL أكثر من فورتران FORTRAN ، وتظهر الفوارق فى البرمجة العلمية أو التطبيقات غير المخصصة لمجال الأعمال .

هناك سبب آخر لفشل لغة PL/1 هو عدم تقديم إصدار خاص للكمبيوتر الشخصى الذى لم يكن يملك ذاكرة كافية أو سعة تخزين كبيرة لمثل تلك اللغة ، وحتى بعد أن أصبحت الكمبيوتر الشخصى قادرا على استيعابها فإن اللغات الأخرى مثل : سى C وفيجوال بيزك Visual Basic احتلت ساحة الكمبيوتر الشخصى ولا تستطيع لغة PL/1 منافستها أو تحقيق نجاح بجانبها .

لغة RPG

هناك لغة أخرى للبرمجة تواجدت لسنوات عديدة هى لغة برنامج مولد التقارير RPG اختصار Report Program Generator ، وتوجد إصدارات أحدث مثل RPGII و RPGIII ، وبدأت لغة RPG كبرنامج يكتب التقارير فقط ، وكان الغرض منها هو استخدامها لغير المبرمجين لإنشاء تقارير من بيانات الملفات .

إن لغة RPG ليست مثل معظم لغات البرمجة الأخرى فاللغات السابقة عبارة عن لغات إجرائية Procedural بمعنى أنها توفر أوامر مستقلة يقوم الكمبيوتر بتنفيذها فى ترتيب متتال حتى تنتهى المهمة ، أما لغة RPG فهى لا تحتوى على أوامر نموذجية

تقليدية ، وتعتبر الفكرة المنطقية بها غير إجرائية Nonprocedural (بعض إصدارات RPG الحديثة توفر قدرات إجرائية محدودة) .

يكتب برنامج لغة RPG باتباع قواعد صارمة لأماكن الأعمدة والكود المكتوب بها لذا سوف يجد غير المبرمجين أو المبرمجون القدامى ممن لم يعتادوا على لغة RPG صعوبة فى فهم برامج لغة RPG ، ومما يجعل الأمور أسوأ وجود عدة إصدارات غير قياسية من اللغة RPG تستخدم على نطاق واسع .

إن وضع الكود فى الأعمدة يجب أن يكون دقيقا للغاية فإذا قمت بإزاحة أى سطر من البرنامج إلى اليمين أو اليسار بضعة مسافات لن يعمل البرنامج ، ويكون برنامج RPG صعب التتبع ، وفى غاية الصعوبة أن تجعله يعمل فى أول مرة تكتبه فيها .

يستخدم مبرمجو RPG ورقة مواصفات Specification Sheets لكتابة برامجهم ، وتشبه هذه الورقة ورقة الرسم البيانى حيث توجد خلايا معنوية لمختلف الأوامر فى لغة RPG ، وباستخدام رقم المواصفات فى كتابة البرنامج سوف يتأكد المبرمجون من صحة أماكن الأعمدة للبرنامج .

كان المبرمجون يستخدمون لغة RPG مبدئيا على الكمبيوتر الصغير Mini ، وبذلك كانت أكثر اللغات استخداما على ذلك النوع (ظهر فقط نوعان من برامج RPG للكمبيوتر الشخصى وتراجعت مبيعاتها نتيجة عدم الإقبال) ، ويتوقع الكثيرون أن تختفى RPG مع اختفاء أخر كمبيوتر صغير Mini .

لغة APL ولغة ADA

هناك أيضا للبرمجة لغة APL ولغة ADA تم استخدامهما خلال السنوات الماضية أولهما (لغة البرمجة A) أو APL (A Programming Language) وهى لغة برمجة رياضية متقدمة طورتها شركة IBM تختلف عن لغتى كوبول COBOL وفورتران FORTRAN فيحتوى برنامج APL على العديد من الرموز الغريبة (المثلثات والأقواس والأسهم وغيرها) ، وتتطلب معدات كمبيوتر خاصة لتوليد هذه الرموز ، ونظرا لبطء سرعتها مقارنة بلغات البرمجة الأخرى فإنها تفقد مكانتها .

لغة أدا ADA مسماة على اسم آدا بايرون ابنة الشاعر لورد بايرون وصديقة شارلز باباج الأب الأكبر للكمبيوتر) وتستخدم فى وزارة الدفاع الأمريكية وبعض الهيئات

الحكومية الأخرى التى حاولت جعل لغة البرمجة قياسية حتى تكون جميع البرامج متجانسة ويعتاد مبرمجو الحكومة على اللغة المستخدمة فى جميع الهيئات ، أما الخبراء فيرون أن ADA هى لغة برمجة متوسطة الأداء وصعبة التعلم .

قد ترغب فى تعلم لغة ADA عند العمل فى شركة تكتب أو تقوم بصيانة البرامج للحكومة الأمريكية ، ونظرا لأن عددا قليلا من العاملين بالحكومة يعرفون لغة ADA يزداد الطلب على مبرمجى ADA خاصة فى تطبيقات الطيران والدفاع والفضاء بالإضافة إلى ذلك فأثناء انتقال الهيئات الحكومية للعمل مع لغات أخرى مثل C++ يزداد الطلب أيضا على المبرمجين الذين يفهمون لغة ADA حيث بإمكانهم القيام بعملية التحويل .

لغات البرمجة الحديثة

قطعت البرمجة شوطا طويلا ، ومع كل لغة برمجة جديدة يأتى وعد بسرعة التعلم ومزيد من القدرة على تعديل التعامل مع الكود ، وسرعة المبرمجين فى التعامل معها واستخدامها .

زاد الاتجاه نحو لغات البرمجة الرسومية وأصبح فى إمكان المبرمجين وغير المبرمجين إنتاج برامج عن طريق تحريك بعض العناصر الرسومية على الشاشة ، ويرجع السبب المباشر فى انتشار تلك اللغات إلى شعبية واجهة المستخدم الرسومية GUIs مثل أنظمة ويندوز وماكنتوش .

أكثر لغات البرمجة انتشارا خلال السنوات القليلة الماضية اللغات التى تعمل على الكمبيوتر الشخصى .

البرمجة الهيكلية ولغة باسكال Pascal

فى سنة ١٩٦٨ تمت كتابة أول مترجم باسكال Pascal التى يرجع اسمها إلى عالم الرياضيات الفرنسى ، وتعتبر لغة البرمجة باسكال Pascal لغة برمجة جيدة ذات غرض عام تدعم العمل العلمى والأعمال ، وتختلف قدرات المدخلات والمخرجات الخاصة بها عن لغات البرمجة الأخرى المستخدمة فى الأعمال مثل كوبول ، وعلى الرغم من ذلك فإن لغة باسكال لغة قوية قادرة على أداء العديد من الأشياء .

من أهم مزايا لغة باسكال Pascal أنها تدعم مفهوم البرمجة الهيكلية جيدا ، وقد تم اكتمال الدعائم الثلاث الأساسية للبرمجة وهى : التتابع Sequence ، والقرار Decision ، و التكرار Looping فى تصميم لغة باسكال .

لا تحتوى برامج باسكال تقريبا على كود متشعب على خلاف البرامج الأخرى فهناك بعض نسخ لغة باسكال لا تحتوى على عبارات GOTO التى تمنح الانتقال غير المشروط الذى توفره اللغات الأخرى .

خلال السبعينيات ساد تفكير حول أن لغة باسكال ستصبح لغة البرمجة الوحيدة لكنها لم تستطع تحقيق هذا الهدف وبدأ استخدامها فى الاضمحلال ، إذ سرعان ما انتهى ذلك فى الثمانينيات فنادرا ما يستخدمها أحد فى الأعمال الهندسية أو الإدارية ، وتستخدم الآن كأداة تعلم للبرمجة وعلى أجهزة المنزل .

هناك تساؤل وجدال حول ضعف لغة باسكال الآن بالنسبة للغات البرمجة على الرغم من بدايتها القوية ، ويرجع السبب فى هذا إلى قوة المنافسة بين لغات سى التى انتشرت فى أوائل الثمانينيات على حساب Pascal .

يعتبر برنامج لغة باسكال نموذجا حرا ، بمعنى أنه يمكن وضع سطور ومساحات فارغة فى البرنامج (مساحة بيضاء) لإمكان قراءة البرنامج بشكل أفضل ، وعادة يقوم المبرمج بعمل سطور مجتمعة من الكود ويضيف سطورا خالية بين أقسام البرنامج لتوضيح أجزاء البرنامج المختلفة والمساعدة فى عملية التعديل .

استمرت شركة بورلاند Borland أكبر مروج للغة باسكال ، وقامت شركة تحت اسم جديد Inprise بتحقيق سمعة كبيرة فى كونها مورد منتجات البرمجة الخاصة بلغة باسكال منذ ظهور نسخة Turbo Pascal1.0 الأولى التى أصبحت باسم Borland Pascal فى إصدار ويندوز حيث تطورت لتصبح بيئة برمجة كامل تشتمل على أشكال ورسوم بالإضافة إلى مكتبات دعم قاعدة البيانات .

بدون دعم بورلاند للغة باسكال لكانت اللغة انتهت بالكامل قبل التسعينيات ، وبفضل شركة بورلاند لا تزال هناك قاعدة صغيرة من مبرمجى باسكال على الرغم من جهود تحسين اللغة التى تعد ذكرى حتى للمبرمجين الذين اقتصروا على استخدامها فى الثمانينيات .

لازالت بعض الكليات والمعاهد تطلب من الطلاب المبرمجين أخذ بعض دروس لغة باسكال فى المنهج الدراسى بسبب دعم البرمجة الهيكلية واكتساب بعض عادات البرمجة الجيدة التى تستمر أثناء الانتقال لأى لغة برمجة أخرى .

قامت شركة Inprise بتوليد لغة برمجة جديدة فى التسعينيات باسم دلفى على أمل أن تحتل المكانة بعد لغة باسكال ، وتعد لغة دلفى Delphi لغة مرئية تمكن المبرمجون من عمل تطبيقات ويندوز بأقل مجهود من البرمجة ، وقد قامت شركة Inprise بتصميم دلفى Delphi لمنافسة فيجوال بيزك Visual Basic لكن استخدامات فيجوال بيزك Visual Basic أكثر بكثير من Delphi .

لغة سى C

لأخذ فكرة عامة عن لغة سى C والمكانة التى احتلتها C وسط باقى لغات البرمجة المستخدمة تم إنشاء لغة C فى معامل بل Bell Labs ، وفى سنة ١٩٧٢ احتاجت معامل بل إلى كتابة نظام تشغيل جديد ، وفى ذلك الوقت كانت أنظمة التشغيل تكتب فى شكل لغة تجميع Assembler لأن لغات البرمجة المتقدمة لم تكن فعالة بدرجة كافية ، وتتمثل صعوبة لغة البرمجة ذات المستوى الأدنى فى أن الكود الخاص بها يصعب تعديله ، ومع تحديث أنظمة التشغيل ظهرت بعض المشكلات والصعوبات للمبرمجين الذين يرغبون فى تحديث هذا الكود بلغة التجميع Assembler .

كان نظام تشغيل يونكس UNIX هو نظام التشغيل الناتج عن هذا المجهود والذى لا يزال يستخدم بكثرة فى الكمبيوترات الصغيرة Minicomputers .

لم ترد معامل Bell أن يكون نظام التشغيل الجديد الخاص بها صعب التعديل كما فى الأنظمة السابقة ، ولم تكن هناك لغة متقدمة فى ذلك الوقت يمكن أن تؤدى المهمة ، لذلك تم الاتفاق على كتابة لغة برمجة جديدة تكون سهلة التعديل كما فى لغات البرمجة المتقدمة الأخرى وتتميز بالفاعلية أيضا كما فى الكود المكتوب بلغة التجميع لتكون النتيجة فى النهاية لغة البرمجة سى C .

يطلق على لغة سى C لغة متقدمة High Low Level Language بمعنى أنها تدعم جميع أساسيات البرمجة الخاصة بأى لغة متقدمة مشتملة على أساسيات البرمجة الهيكلية ، وتقوم أيضا بالترجمة إلى كود محكوم وفعال يعمل تقريبا فى سرعة لغة

التعديل والتحديث :

انتشرت شعبية C بدرجة كبيرة فقد أحبت الشركات فكرة الحصول على المزيد من البرامج الفعالة ، ويمكن لبرنامج C أن يعمل بسرعة عشرة أضعاف أى برنامج كوبول مماثل ، وعلى الرغم من أن C لم تحتل مكان كوبول فإن هذا المفهوم بدأ ينتشر فى العديد من قطاعات البرمجة .

تعد لغة البرمجة C هى اللغة التى يريد معظم الأشخاص تعلمها ولم تشهد صناعة الكمبيوتر مثل هذا الانتشار والقبول والاتجاه نحو لغة برمجة أكثر من مما حدث معها فى أواخر الثمانينيات وأوائل التسعينيات ، وواجهت المدارس أعدادا ضخمة من المسجلين فى فصول لغة C واستمرت مبيعات كتب C فى تحقيق أرقام قياسية ، واتجهت معظم الشركات إلى تحويل جميع إدارات البرمجة لديها إلى استخدام C أو C++ وازداد الطلب على مبرمجى C أو C++ فى إعلانات الوظائف .

يتم كتابة معظم البرامج المستخدمة حاليا بواسطة لغة C أو C++ التى كانت تكتب فى الماضى باستخدام لغة التجميع ، كما يتم كتابة برامج أوراق العمل الشائعة ومعالجات النصوص عن طريق C لضمان الحصول على مزيد من الفاعلية وقابلية التعديل .

لا تعد C الاختيار الجيد كأول لغة كمبيوتر تتعلمها لذلك يمكن أن تكون لغة C هى خطوة تالية بعد هذا .

بغض النظر عن أهداف البرمجة الخاصة بالمبرمج فيجب أن تكون لغة C لغة سيتعلمها فى النهاية .

يعتبر البعض لغة C لغة محيرة ، ويمكن أن تكون كذلك إذا لم تكتب وترتب جيدا ، وتعد لغة ذات نموذج حر تسمح بإضافة تعليقات حتى يمكن شرح ما يدور داخل البرنامج ، ويساعد هذا على تبسيط برامجها .

يطلق على C لغة البرمجة "المكتوبة للمبرمجين بواسطة المبرمجين" ، وتحتوى العديد من لغات البرمجة على قواعد تقيد المبرمج بعكس لغة C التى توفر الانطلاق للمبرمج بقدر أكبر مما تمنحه لغات البرمجة الأخرى .

من ناحية أخرى يجب أن تكون عين المبرمج دقيقة للأخطاء المنطقية لأن مترجم لغة C يكشف عن أخطاء قليلة مما يجعل المبرمج مسئولا أكثر عن ضمان دقة الكود

المكتوب .

تملك لغة C كلمات قليلة ومعاملات متعددة فاللغة C تحتوى على ٣٢ أمرا فقط بعكس اللغات الأخرى مثل كوبول وبيزك QBASIC التى تملك فوق المائة أمر ولذلك فإن لغة C تعتبر أكثر إنجازا .

تمتلك لغة سى C معاملات أكثر من اى لغة برمجة أخرى باستثناء لغة APL العلمية ، والمعامل Operator هو رمز معين يؤدى عملية معينة على البيانات ، وليست كل المعاملات حسابية ، لكن تعد المعاملات الحسابية الأولية هى الطريقة الواضحة لتعلم تلك المعاملات ، وتستخدم معظم لغات البرمجة المعاملات التالية :

معامل	مثال	شرح
+	٨+٧	يقوم بالإضافة (الجمع)
-	٣-٨	" بالطرح
*	٩*٥	" بالضرب
/	٦/٢٤	" بالقسمة

على خلاف اللغات الأخرى فإن تعلم عمل معاملات لغة C يعتبر مهما للغاية فهى تملك عددا كبيرا من المعاملات التى تجعل الأشخاص يعتقدون أنها لغة غامضة ، والعديد من معاملاتها يأخذ مكان الأوامر المستخدمة بواسطة لغات أخرى وتؤدى استخدامات المعاملات فى لغة C إلى زيادة فاعليتها وجعلها أكثر إيجازا .

لغة سى بلاس C++

لغة سى بلاس C++ هى نسخة معدلة من لغة سى C تم تصميمها فى أوائل الثمانينيات تتميز عن باقى اللغات الأخرى أنها تأسست على لغة C فمعظم برامجها تبدو مماثلة لبرامج لغة سى ، وتمنح لغة C++ بعض الأوامر والمعاملات الإضافية .

يرجع السبب الأساسى لنجاح C++ أنها تحتوى على قدرات البرمجة التى تعتمد على توجيه الكائنات ، وتعد برمجة توجيه الكائنات (البرمجة الشيئية) Object-Oriented Programming (OOP) طريقة مختلفة لكتابة البرامج .

تعتبر لغة C++ أكثر تشددا من لغة C العادية فلغة C++ لا تعطى المبرمج الحرية التى تمنحها له لغة سى C لكنها فى نفس الوقت لا تسمح للأخطاء الخفية بالدخول فى

الكود .

لغة بيزك BASIC

ظهرت لغة بيزك BASIC فى كلية Dartmouth ، والكلمة اختصار (شفرة التعليمات الرمزية لكافة الأغراض للمبتدئين) Beginner's All-purpose Symbolic Instruction Code .

تعمل لغة بيزك BASIC فى بيئة التفسير Interpreted على الرغم من أن النسخ الحديثة BASIC قابلة للترجمة compiled ، وعن طريق التفسير يمكن أن يركز المبتدئون على لغة البرمجة ولا يصيبهم القلق بشأن تفاصيل ترجمة Compiling البرنامج .

مثل أى لغة مفسرة Interpreted تعمل برامج لغة بيزك BASIC المفسرة بسرعة أقل من البرامج المترجمة ، لذلك لا تستخدم برامج بيزك BASIC المفسرة كثيرا فى الأعمال .

على مدار السنوات تم توزيع BASIC فى عدة نماذج مختلفة ، فلغة بيزك BASIC الأصلية لديها هيكل صغير للغاية ، وتحتوى على مجموعة جامدة من قواعد التشفير . لغة BASIC لم تتم هيكلتها بشكل جيد ، ويمكن أن يصبح البرنامج متشعبا بصورة كبيرة ، كما يحتاج كل سطر إلى رقم ، وقد كانت الممارسة الشائعة هى جعل أرقام الأسطر بمضاعفات الرقم عشرة حتى يمكن إضافة تسعة سطور بين السطور الحالية عند الحاجة .

على الرغم من أن لغة بيزك BASIC وجدت خصيصا للمبتدئين ، ولا تزال مفيدة لتعريف أساسيات البرمجة فالنسخ الحديثة اليوم منها تنافس لغات باسكال وسى عن طريق توفير مجموعة غنية من عناصر البرمجة الهيكلية ، فيما يلى أسماء إصدارات مختلفة من لغة بيزك BASIC وتطوراتها :

* BASICA (بيزك متقدم) BASIC Advanced .
* (GW BASIC) .
* Power BASIC .

* Turbo BASIC .
* QUICK BASIC (BASIC مترجم) .
* QBASIC (بدأت مع الإصدار 5.0 من MS-DOS) .
* Visual Basic (ظهرت نسخة Dos ونسخة ويندوز بنسخ مختلفة) .

يعتبر تبنى مايكروسوفت للغة بيزك من أهم عوامل نجاح اللغة ، وقدمت منها نسخا عديدة بالتحسينات على اللغة فى كل نسخة تصدرها آخرها فيجوال بيزك Visual Basic التى هى عبارة عن أداة برمجة رسومية تستطيع من خلالها كتابة برامج ويندوز تخصصية .

باستخدام بيزك فى الإصدار Quick basic انتقلت اللغة من مرتبة الهواة إلى فئة جديدة تم كتابتها وهيكليتها وترجمتها بعناية ودقة ، ومن ثم دخلت حيز المنافسة .

بظهور مترجم بيزك (Quick BASIC Compiler) انتهى الأمر إلى وجود مترجم متكامل للشاشة بأكملها ، واشتملت نسخ بيزك BASIC على محرر سطور أو محرر محدود للشاشة .

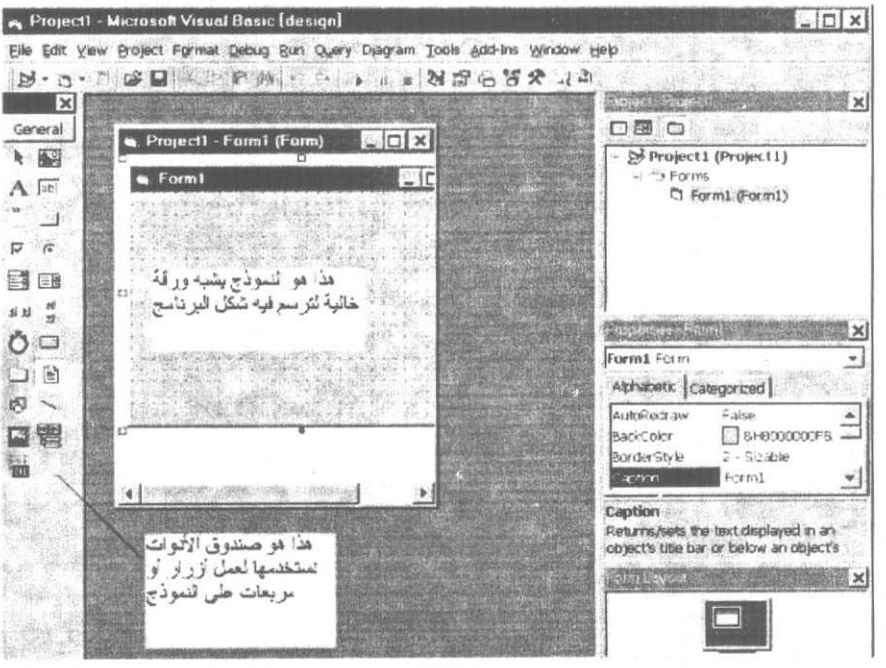
تعد لغة Qbasic لغة الاختيار لأغلب المبرمجين المبتدئين ، ويعتبر دعم البيانات النصية من أقوى مقومات بيزك BASIC وتتكون البيانات النصية من حروف وكلمات وجمل فى مقابل البيانات الرقمية ، وتحتوى لغة بيزك على بعض الوظائف المدمجة مثل : المحاذاة يسارا والمحاذاة يمينا للنصوص بالإضافة إلى التقاط حرف أو أكثر من النصوص والوظائف المدمجة وتعتبر قدرات التعامل مع النصوص ميزة هامة تتفوق بها على لغات سى C وباسكال Pascal وكوبول COBOL أو فورتران FORTRAN .

فيجوال بيزك والبرمجة الرسومية Visual Basic

تعتبر لغة البرمجة فيجوال بيزك Visual Basic أكثر اللغات تميزا ، وتشتمل على مجموعة من متحكمات رسومية تستطيع من خلالها بناء برامج بيزك المترجمة Compiled أو المفسرة interpreted .

عند تشغيل فيجوال بيزك Visual Basic لأول مرة ترى شاشة يبدو منها مصطلح المرئى Visual وصفا دقيقا حيث تفتح شاشة التشغيل الأولى الإطارات اللازمة

للبرمجة الرسومية .



تصعب كتابة برامج ويندوز لكل من المبرمجين المبتدئين والمتقدمين ، وقد كانت كتابة برنامج ويندوز للمبتدئين عبئا إلى أن جاءت لغة فيجوال بيزك لتعتنى بجميع تفاصيل ويندوز من أجل أن يتمكن المستخدم من تغيير حجم إطار والاختيار من القائمة والنقر بالفأرة أو ضغط مفتاح ، وتعتبر اللغة من أولى لغات البرمجة فى بيئة ويندوز التى توفر كلا من المترجم Compiler والمفسر Interpreter معا .

يمكن تعلم Visual Basic ، واختبار البرامج فى وضع التفسير ، ثم ترجمة البرنامج الذى تم اختباره لتوزيعه بعد ذلك على المستخدمين ، وبالإضافة إلى برامج ويندوز التى تستقر فى الذاكرة أثناء إجراء العمليات الأخرى فإن حفظ السجلات الذى تؤديه ويندوز باستمرار يمكن أن يبطئ من برنامج فيجوال بيزك ، وتقوم ويندوز بمراقبة أنشطة عديدة لم تكن موجودة من قبل فى برامج دوس DOS .

تعد لغة فيجوال بيزك Visual Basic أداة برمجة ويندوز الأولى التى مكنت المبتدئين من إنشاء برامج ويندوز بدون الاستعانة بلغة C التى كانت مستخدمة قبل ذلك باستخدام حزمة تطوير البرامج SDK .

انقسمت بعض آراء البرمجة حول استخدامات فيجوال بيزك وفيجوال سى مقابل لغة أكثر جدية مثل C++ ، نظرا لقيامها بأداء معظم العمل للمبرمج فالمعارضة تنص على أن المبرمج لديه قدرة تحكم أقل على البرنامج النهائى على عكس استخدام لغة أخرى مثل C++ فى كتابة البرنامج .

من الأفضل عدم الاعتقاد بأن بيئة برمجة فيجوال بيزك موجهة للكائنات Object Oriented ، وذلك على الرغم من أن هذه اللغة تعمل مع الكائنات على الشاشة لكنها تعتبر لغة الاتجاه نحو الرسوم والأحداث وليس الكائنات .

برمجة الألعاب

إذا تجولت بين متاجر الكمبيوتر للبحث عن نسخ برامج دوس فلن تجد شيئا بسبب أن معظم الأجهزة اليوم تعمل على ويندوز ، والاستثناء هنا هو الألعاب ، فالعديد من الألعاب الشائعة لا تعمل تحت بيئة ويندوز بل تشترط بيئة دوس لتشغيلها .

تتمثل مشكلة الألعاب فى احتياجها الشديد لموارد الكمبيوتر فهى لا تحتاج فقط إلى المتطلبات المعتادة للبرمجة مثل معالجة الأرقام وملفات القرص أو تفاعل المستخدم بل تحتاج أيضا إلى بطاقة صوت وبطاقة فيديو ومكونات الجهاز ، وعلى الرغم من ازدياد سرعة الجهاز عن ذى قبل ، وتوفير واجهات جديدة للبرمجة للوسائط المتعددة فى ويندوز لكن يميل مبرمجو الألعاب إلى نسيان ويندوز ، والاتجاه إلى الكتابة باستخدام لغة الآلة أو يقوم معظم صانعى الألعاب بكتابة الألعاب باستخدام لغتى C و C++ أو لغة التجميع لتحقيق الفعالية التى تحتاجها الألعاب .

قواعد لغات البرمجة

تحتوى كل لغة من لغات البرمجة على مجموعة من القواعد التى تحكم التعامل بهذه اللغة ، وهى قواعد Syntax الطريقة المستخدمة لكتابة أوامر (جمل أو عبارات) لغة البرمجة ليمكن تنفيذها بواسطة الكمبيوتر على النحو الصحيح .

تشبه قواعد لغات البرمجة القواعد النحوية فى اللغات البشرية ، حيث يجب استخدام الكلمات المناسبة بالترتيب الصحيح والتنظيم المقيد ليمكن تنفيذ المهمة .

الميزة المهمة فى قواعد لغات البرمجة أنها متشابهة فى المهام ، وليست معقدة

الاستخدام ، وليست كثيرة العدد كما هو الحال فى قواعد اللغات البشرية فقواعد لغات البرمجة أكثر سهولة وأقل عددا وأشد بساطة ، وتمتلك منهجية متشابهة فى معظم لغات البرمجة .

تتكون معظم أوامر (عبارات أو جمل) البرمجة من كلمة أو كلمتين رئيسيتين ، متبوعة بوسيط (عامل تشغيل) أو أكثر (لتنفيذ المهمة عليه) ، مع معامل أو أكثر (لتحديد أسلوب تنفيذ المهمة) .

قد تتكون أوامر (عبارات أو جمل) البرمجة من كلمة فقط ، أو من كلمتين ، وقد يكون الوسيط عدديا أو حرفيا ، وقد يكون المعامل أداة تعديل لتحديد سياق تنفيذ المهمة .

من أفضل وسائل تعلم قواعد لغة البرمجة : مراجعة دليل استخدام هذه اللغة الذى يتوفر ضمن اللغة نفسها على هيئة كتيب مرفق ، والاطلاع على تعليمات Help المساعدة الفورية ضمن اللغة ، أو قراءة دليل استخدام متوفر على موقع اللغة فى شبكة الإنترنت ، أو تجربة وفهم نماذج البرامج المرفقة فى تعليمات اللغة .

للغات البرمجة قواعد مختلفة (قد تتشابه) لكنها لا تتماثل ، وقد يكون الاختلاف بينها بسيطا ، وقد يكون الاختلاف كبيرا إلا أن الهيكل العام والمنهجية متشابهان تشابها كبيرا ، وتقوم قواعد ومنهجية لغات البرمجة بأداء نفس المهام تقريبا مثل :

- **مهام الإدخال والإخراج Input and output :**

تختص جمل أو طرق الإدخال والإخراج بإدخال البيانات إلى جهاز الكمبيوتر من وسيط إدخال ، واستقبال هذه البيانات فى مستودع ما (قد يكون جزءا من الذاكرة مثل إدخال أرقام للحساب) حيث تتم معالجتها بواسطة الكمبيوتر ضمن مهام المعالجة Processing التى يقوم بها البرنامج ، ثم يتولى البرنامج دفعها إلى وسيط إخراج (شاشة أو طابعة أو قرص أو مكبر صوت) لإخراج المعلومات .

تستخدم لغات البرمجة كلمات للتعبير عن الإدخال ، وكلمات أو جمل أو وسائل للإخراج إلى الوسيط المناسب ، كما تستخدم لغات البرمجة أساليب وطرقا معينة لكتابة جمل المعالجة وحفظ ناتج المعالجة .

- **مهام التكرار Looping :**

عندما تحتاج إلى تكرار مهمة ما ولتكن كمثال حساب مرتبات الموظفين العاملين فى شركة من الشركات أو مؤسسة ما من المؤسسات فإن ما يجرى فى الحقيقة هو تكرار تـــنفيذ المهمة لكافة الموظفين العاملين فى تلك المؤسسة بناء على الأسس الموضوعة لحساب الأجور التى تتكون من المرتب الأساسى مضافا إليه الحوافز بناء على أسس العمـــل ، ومخصـــوما منه الجزاءات والخصومات المقررة بناء على سياسة الأجور المحددة ، لذلك فإن وضع البرنامج يتطلب تكرار العملية لكل موظف ، وتحتوى كافة لغات البرمجة على طرق للتكرار لتنفيذ المهام المتكررة .

- **مهام التفرع Branching ومهام الشروط Conditions :**

لاتخـــاذ قـــرار ما أو للتفرع إلى أداء مهمة ما فإن لغات البرمجة تحتوى على اتخاذ القـــرار بناء على شروط معينة ، سواء أكان هذا التفرع تاما أو مرتدا ، فالتفرع التام يعـــنى الذهاب بلا عودة ، أما التفرع المرتد فقد يعود بقيمة ما ، أو قد يعود بعد تنفيذ مهمة ما .

لنضرب مثالا على الحالتين ففى حالة التفرع التام لنفرض قيام مبرمج بكتابة برنامج لحساب مرتب كل موظف من مجموعة عاملين بشركة ما ، فقد يبدأ حساب المرتبات بإدخال أسماء الموظفين ، أو قراءة بيانات الموظفين من ملف قاعدة بيانات ، ويكرر تنفيذ المهمة لكل موظف ويطبع النتيجة على الورق حتى ينتهى بآخر موظف ، وعند الوصول إلى هذه الحالة فإنه يجعل البرنامج يتفرع إلى نهايته .

فى حالـــة الـــتفرع المـــرتد فى نفس المثال فإنه قد يجعل برنامجا فرعيا يقرأ بيانات الموظـــف من ملف أو يستقبل مدخلات البيانات من لوحة المفاتيح ثم يعود بالبرنامج إلى (جزء حسابِ المرتب فى البرنامج) ناقلا البيانات المدخلة إليه ليقوم بالحساب بناء عـــليها ثـــم يكرر المهمة مرة أخرى بإعادة استقبال بيانات الموظف التالى ، وحساب مرتبه ، وهكذا دواليك حتى تنتهى عملية حساب المرتبات لكل الموظفين .

مما سبق يتضح أنه كما سبق القول فإن لغات البرمجة تمتلك قواعد بسيطة متشابهة ، وكمثال على ذلك تأمل الجملة التالية فى لغة بيزك :

A=A+74

إن نفس هذه الجملة فى لغة البرمجة سى هى :

```
A=A+74;
```

فالجملـــتان متشابهتان شكلا وأداء ، تعنى الجملة الأولى إضافة رقم ٧٤ إلى الموجود فى مخزن المتغير A ، ووضع الناتج فى نفس مخزن المتغير A ، والاختلاف الوحيد بيـــن الجملتين هو : وضع الفاصلة المنقوطة عند نهاية جملة لغة البرمجة سى ، فهذه هى قواعد لغة البرمجة سى ، وكمثال آخر فالجملة التالية فى لغة بيزك :

```
A=3+17
```

هى نفس الجملة التالية فى لغة البرمجة سى :

```
A=3+17;
```

ومثيلتها فى لغة البرمجة كوبول هى :

```
COMPUTE a_var=3+17
END - COMPUTE
```

بهـــذا يكـــون واضـــحا أنه بالرغم من أن المهمة هى نفسها لكن تختلف صيغة جملة الأوامر فى التركيب والنص .

اللغات الأفضل

يصعب الإجابة على سؤال عن أفضل لغة برمجة بين اللغات العديدة ، فاختيار أفضل لغة يعتمد على المهمة .

لمعرفة أفضل لغة للتعلم أولا تعتبر لغة بيزك BASIC (أو مشتقاتها مثل QBASIC) أفضل لغة للمبرمجين المبتدئين لتنمية مهارات البرمجة .

يحتاج معظم الأشخاص إلى قدرة تحكم أكثر من التى تدعمها لغة بيزك Qbasic فبمجرد تمكنهم من لغة تمهيدية يتجه معظم الأشخاص إلى فيجوال بيزك Visual Basic ولغة سى C أو C++ .

تتطلب الوظائف الحالية معرفة لغة سى C أو C++ أو فيوال بيزك وكوبول فى الأقسام التى تستخدم كمبيوتر كبير Mainframe ، وللعمل بالبرمجة فى منظمات كبيرة تعتمد على كمبيوتر كبير يجب تعلم لغة كوبول COBOL .

عامة فمعظم الأشخاص يمكنهم التعامل مع الكمبيوتر الشخصى ، ويسهل تعلم البرمجة باستخدام الكمبيوتر الشخصى لأنه يمنح نتائج سريعة ، ويمكن تعلم البرمجة فى

المنزل .

إن لغة C ولغة C++ هى أكثر اللغات رواجا ، كما أن فيجوال بيزك لها تأثير فعال أيضا .

يتم تحديد اللغة التى تبرمج بها على أساس اللغات التى تعرفها فإذا كنت تعمل على مشروع عاجل (كما فى أغلب مشاريع معالجة البيانات) فلن يكون لديك وقت لإتقان لغة جديدة ، لذلك فإذا كانت فيجوال بيزك Visual Basic هى كل ما تعرفه فهذه هى اللغة التى سوف تستخدمها .

من الصعب تحديد لغات البرمجة التى يجب على المبرمج معرفتها ، وهناك بعض المبرمجين ذوى المهارات العالية والأجر المرتفع يعرفون لغة واحدة فقط باحتراف عال ، لكن أغلب المبرمجين يعرفون أكثر من لغة .

فى النهاية تجد نفسك تفضل لغة عن الأخرى ، وتتقن لغة واحدة فقط على الرغم من تعلم أكثر من لغة .

إن اللغة التى تستخدمها أى شركة فى كتابة تطبيق معين يتم تحديدها على أساس عدة عوامل ، فالشركات قد يكون لديها مترجم Compiler أو أكثر ، وإذا كان الأمر كذلك فستتم كتابة التطبيقات باستخدام واحد منها ، وقد يكلف مترجم الكمبيوتر الكبير آلاف الدولارات لذلك يصعب على الشركات الانتقال من لغة إلى أخرى باحثة عن الأفضل ، وعادة فاللغة المتداولة حاليا هى التى تستخدم فى العمل .

يلعب التطبيق دورا كبيرا فى تحديد أفضل لغة للاستخدام ، ويتم عمل التطبيقات العلمية على الكمبيوتر الكبير بصورة أفضل بلغة فورتران FORTRAN ، وبالنسبة لمعالجة كميات ضخمة من البيانات الانتقالية فإن لغة كوبول COBOL ستكون اللغة المختارة .

الشىء الوحيد الذى يجب تذكره بشأن لغات البرمجة الحديثة هو أن جميعها تعتبر ذات طبيعة صالحة لكتابة أى تطبيق .

بغض النظر عما ستقوم باستخدامه فى النهاية فإن مجال البرمجة ممتع ومجز كذلك مما يدعو للانضمام لمرتبة المبرمجين وكتابة البرامج .

خلاصة

إن فهم طريقة العمل الداخلية للكمبيوتر مطلب أساسى للتميز والتفوق فعند تعلم ما يحدث بداخل الصندوق تنمى مهارات البرمجة .

خطت البرمجة خطوات واسعة منذ الأيام الأولى حيث توصيل الأسلاك والمفاتيح ثم الوصول إلى لغات البرمجة عالية المستوى مثل فورتران .

أهمية فهم البت bit والبايت byte أن على المبرمجين العاملين فى مجال الاتصالات من فهم البت والبايت حتى يتمكنوا من التعامل معها خاصة فى مجال الاتصالات .

إن البرمجة للإنترنت هى إحدى أفضل الوظائف اليوم كذلك فإن تقنية الاتصالات العالمية تتوسع باستمرار وفى تحسن سريع ومتواصل ، وتتطلب دائما مبرمجين للمعدات الجديدة والكمبيوتر التى تسهل عملية الاتصالات ، ولهذا يجب على المبرمج فهم التفاصيل الدقيقة للكمبيوتر مثل البت والبايت .

أصبحت لغات البرمجة أسهل فى الاستخدام ، وتحتوى معظم لغات البرمجة اليوم على بيئات متكاملة كما تمنح وتوفر دعم البرمجة الهيكلية .

تتطلب واجهات المستخدم الرسومية مثل ويندوز المزيد من أدوات البرمجة المعقدة التى توفرها لغات سهلة الاستخدام مثل فيجوال بيزك Visual Basic .

تتغير شعبية لغات البرمجة على مدار الوقت ، وقد احتلت لغة C ولغة C++ الصدارة فى عالم البرمجة ، ومن المتعارف عليه أن C++ تعد لغة أكثر قوة بالرغم من إمكان كتابة أى برنامج ويندوز باستخدام فيجوال بيزك Visual Basic فإن لغة C++ تمنح مرونة وتكاملا أكثر ، وبواسطة مترجم C++ الخاص بويندوز مثل Visual C++ يتمكن المبرمج من إنشاء أكثر برامج ويندوز .

تعد لغة C++ أصعب فى تعلمها من فيجوال بيزك Visual Basic وبالإضافة إلى ذلك يصعب تعديلها فى أوقات كثيرة .

مبادئ برمجة الإنترنت

أصبحت الإنترنت جزءا حيويا من جميع الأعمال والأنشطة والاتصالات ، وتتطور كلما مر الوقت لتصبح أكثر نشاطا واتساع خدمات وتوفر وظائف ووجود أخبار وتحليلات فورية وبث إذاعات وحديث هاتف .

تتطلب الإنترنت برامج ، وهناك طرق عديدة لكتابة تطبيقات الإنترنت التى تتغير تقنية البرمجة لها بسرعة عالية .

لكى يصبح المبرمج متمشيا مع تطور برمجة الإنترنت عليه إتقان مصطلحاتها وفهم طريقة عملها ، فبرمجة الإنترنت هامة ، ويجب معرفة تعامل الإنترنت مع اتصالات الحاسبات ، وانتقال البيانات عبر الإنترنت حتى تجد وجهتها فى محطة الوصول ، واستخدام متصفحات الإنترنت وطبيعتها ، واستخدام لغة النص المتشعب HTML لإنشاء صفحات ويب وتنسيقها ، ومدى أهمية لغات النصوص المكتوبة Script فى تقنية الإنترنت ، واستخدام وإضافة عنصر ActiveX فى تطبيق الإنترنت .

اعتبارات البرمجة للإنترنت

تعمل الطبيعة المتغيرة للإنترنت على جعل مفرداتها قديمة بمجرد تعلمها أحيانا ، لكن أساس الإنترنت لا يتغير فاتصال الحاسبات والعنونة وتوزيع الحزم بقيت وسوف تستمر لبعض الوقت ، أما الطبقة القابلة للبرمجة من الإنترنت فتتعرض للتغير بسرعة ويعتبر فهم الإنترنت أمرا حيويا لأى شخص يرغب فى تعلم تقنية برمجة الإنترنت .

بعد فهم مصطلحات الإنترنت يصبح سهلا فهم تقنية برمجة تطبيقات الإنترنت ، فالإنترنت ليست مجرد شبكة اتصال بل هى شبكة الشبكات ، وأهم عناصر اتصالات الإنترنت هو العثور على الكمبيوتر الصحيح فى الجانب الآخر لتوصيل الرسائل .

نظرا لأن اتصالات الحاسبات هى هدف الإنترنت فإن الاتصالات تستخدم الأرقام الثنائية من الآحاد والأصفار (1 و 0) فى اتصالات الإنترنت ، ولحسن الحظ فإن لغات برمجة تطبيقات الإنترنت تخفى الأساس الثنائى لبيانات اتصالات الإنترنت .

هناك طرق عديدة للاتصال بالإنترنت ومازالت اجهزة المودم هى الوسيلة الأساسية للمستخدمين ، وتقوم بعض الشركات بتثبيت وسائل اتصال عالية السرعة بالإنترنت ، ويستقبل بعض المستخدمين إشارات أطباق الاستقبال ذات السرعة العالية ، وكلما كان الاتصال أسرع كلما تمكن المستخدم من إرسال بيانات الإنترنت بصورة أسرع .

لن تحتاج تطبيقات الإنترنت التمييز بين طرق اتصال الكمبيوتر مع الإنترنت فوسيلة الاتصال ونوعه أمر غير هام ، لكن يجب تذكر عدم قدرة كل المستخدمين على

الوصول السريع مع الإنترنت ، وعلى ذلك فعند تطوير تطبيق يجب جعله على درجة عالية من الكفاءة ، فالرسوم والفيديو قد تأخذ وقتا فى التحميل إذا كان الاتصال بطيئا ويجب على تطبيقات الإنترنت أن تحسن من أداء صفحات ويب ، وليس دفع المستخدمين بعيدا عن مشاهدتها نظرا لطول وقت التحميل .

طريقة استقبال الكمبيوتر لإشارة كمبيوتر آخر بصورة صحيحة تكون عبر اتصال من نقطة إلى نقطة Point to Point التى تسمى عادة اتصال بروتوكول نقطة لنقطة PPP وتوجد تحويلات Switches عبر مسارات الاتصال بين نقاط الاتصال فى الإنترنت ويجب إعداد التركيبة المضبوطة الصحيحة من تلك التحويلات عبر مسار البيانات بين الحاسبات على الإنترنت عندما يكون على الإشارة أن تنتقل من جهاز لآخر .

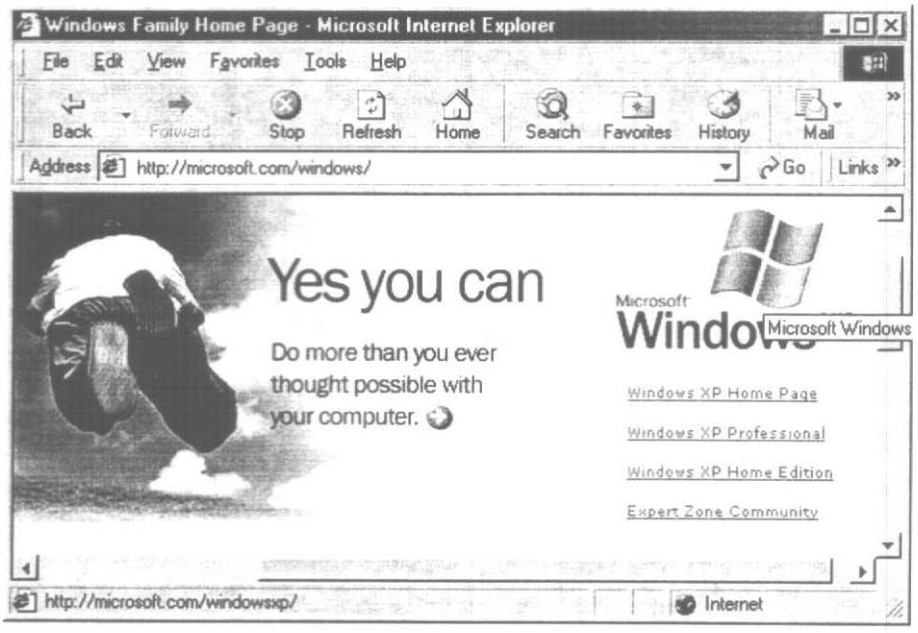
يستخدم اتصال نوع نقطة لنقطة PPP لإرسال البريد الإلكترونى واستقباله ، وربط الاتصال بين المعلومات على الإنترنت إيابا وذهابا بين الحاسبات فصفحات ويب وبرامج الإنترنت نفسها عليها أن تنتقل عبر مجموعة مناسبة من التحويلات قبل أن تظهر على الكمبيوتر الذى سوف يستقبل الإشارة .

عندما يتصل كمبيوتر مرتبط بالإنترنت مع كمبيوتر آخر لايتم إرسال البيانات دفعة واحدة لكن بدلا من ذلك فالخادم يقوم بتقسيم الرسالة إلى حزم Packets قصيرة تتيح للكمبيوتر المستقبل جمعها ثم يرسل ما يفيد استقبالها قبل أن يقوم الكمبيوتر الخادم بإرسال الجزء التالى ، ويكون تصحيح الأخطاء أسهل هنا فالإرسال الخاطئ لأحد الأجزاء يمكن تجاوزه بإعادة إرسال الجزء قبل مرور وقت طويل .

التحدى الأكبر هو أن يتلقى المستقبل الصحيح ما هو مرسل إليه عبر عنوان الحزمة Packet Address فلكل كمبيوتر متصل بالإنترنت عنوان فريد خاص فإذا تلقى كمبيوتر جزءا من البيانات ولم يتطابق عنوان الجزء مع عنوان الكمبيوتر فسوف يرسل الكمبيوتر البيانات حتى يتطابق عنوانها مع ذلك الخاص بالكمبيوتر المعنى .

عند استكشاف شبكة ويب تشاهد صفحات رئيسية Home Pages وهى إطارات رسومية ذات بقع ساخنة قابلة للنقر عليها للانتقال بين صفحات ويب فى الإنترنت ، ليقوم برنامج تصفح ويب بالانتقال خلال نسيج العنكبوت من الاتصالات والعلاقات المتشابكة .

وتكون أول صفحة تقوم بزيارتها فى الموقع هى صفحته الرئيسة Home Page التى تحتوى عادة على روابط Links إلى صفحات أخرى بالموقع أو صفحات رئيسية أخرى فى الإنترنت .



إن كل صفحة ويب موجودة فى مكان ما على كمبيوتر خاص فكيف يتعرف متصفح ويب على الصفحة؟ يحتوى كل موقع ويب على عنوان فريد محدد الموارد Uniform Resource Locator (URL) أو "باحث الموارد الموحد" ، ونقر الارتباط يرشد متصفحة ويب إلى عنوان URL بمعنى أن التجول فى ويب هو تقليب صفحات .

يعمل ويب بصورة متشابهة لنظام النص المتشعب المرتبط الخاص بتعليمات ويندوز Windows فعندما تنقر على إحدى الروابط تنتقل إلى مكان آخر .

إن عناوين URL لصفحات ويب ليست هى نفسها عناوين الحاسبات على الإنترنت المذكورة سابقا فيمكن لكمبيوتر بعنوان إنترنت معين أن يحتوى على عدة صفحات ويب يقدمها للآخرين ، وتبدأ عناوين صفحات ويب عادة بالجزء http:// مثل العنوان http://www.microsoft.com على الرغم أن هذا الجزء لم يعد مطلوبا كتابته فى متصفح ويب .

لغة النص الفائق المتشعب HTML

الإنترنت هى مجموعة حاسبات متشابكة ومتصلة حول العالم ، وليست شبكة ويب هى الإنترنت بل هى فقط نظام إدارة ربط متحركة بين مواقع ويب ، وتعمل على توحيد البحث والتجول .

قد تحتوى صفحة ويب على نصوص أو رسوم أو رموز تتضمن بقعا حية ينقر عليها المستخدم لتأخذه إلى أماكن أخرى ، فأحد أهداف مصمم صفحات ويب هو جعلها تظهر موحدة بغض النظر عن نوع الكمبيوتر المستخدم أو نظام التشغيل ، وتمثل لغة HTML حالة الملف المستقلة عن الجهاز التى يستخدمها المبرمجون لتصميم صفحات ويب ، وهى عبارة عن مجموعة أوامر نصية ينتج بواسطتها متصفح ويب صفحة ويب التى يريدها المصمم .

لا تحتوى تلك الأوامر على نص منسق لكنها تقدم تعليمات التنسيق التى يجب على متصفح ويب أن يتبعها ، وعلى ذلك فعندما يرى برنامج المتصفح أمر توسيط النص <CENTER> سوف يعرف أن عليه توسيط النص التالى لهذا الأمر حتى نهاية الأمر </CENTER> الذى يعبر عن نهاية تنفيذ أمر التوسيط .

عند التجول خلال صفحة ويب فإن الكمبيوتر الخادم البعيد يرسل إلى المتصفح نص لغة HTML فقط ، ويستجيب المتصفح للأوامر بتنسيق النص بالهيئة المطلوبة ، ثم يضع الروابط Links والصور والرسوم حيث يجب أن تظهر ، يستقبل المتصفح أولا صفحة HTML الكاملة يليها الرسوم ومحتوى الوسائط المتعددة الضرورية لاستكمال الصفحة .

فى كثير من الأحيان لن تحتاج إلى استخدام لغة HTML لإنتاج صفحة ويب حيث إن العديد من وسائل تصميم ويب تتيح إبداع صفحات ويب واستخراج كود HTML للصفحة مثل برنامج Front Page الذى يسمح بسحب العناصر من مربع أدوات إلى الشاشة وكتابة نص بداخل المربعات لإنشاء صفحة ويب ثم يترجم الصفحة إلى أوامر HTML ، كما تقوم كل برامج المكتب Office أيضا بحفظ المستندات المحتوية على النصوص والرسوم والجداول والوسائط المتعددة على هيئة صفحات HTML التى

تترجم إلى كود HTML .

يضيف المطورون الجديد باستمرار إلى لغة HTML ، وكلما تصبح ويب معقدة تزداد متطلبات لغة HTML فمثلا : لم تدعم لغة HTML الجداول منذ البداية لكنها الآن شيئا قياسيا عاديا ، إحدى أهم الإضافات هى قدرة صفحة ويب على التفاعل باستخدام برنامج مدمج .

على الرغم من وجود برامج كثيرة لإنشاء صفحات ويب باستخدام الوسائل الحديثة الرسومية للقص واللصق فمازال بالإمكان إنشاء صفحة ويب باستخدام محرر نصوص ومعرفة للغة HTML ، وغالبا ما يستخدم المبرمجون الأدوات المرئية لوضع تخطيط عام لصفحة ويب ككل ، ثم يستخدمون محرر النصوص لكتابة كود HTML وإنهاء العمل بالصفحة .

عنصر التحكم ActiveX

منذ سنوات كان على مصممى صفحات ويب كتابة برامج للتحكم فى صفحات ويب بلغة سى C لدمج تقنية تصفح ويب فى التطبيقات ، ثم ظهرت عناصر التحكم ActiveX المعتمدة على ويب فى تطبيقات الإنترنت .

عنصر التحكم ActiveX برنامج صغير يمكن دمجه بداخل قارئ تحكم ActiveX ، وتدعم العديد من تطبيقات ويندوز Windows تقنية ActiveX حتى يمكن عمل امتداد للتطبيق بإضافة عنصر تحكم ActiveX إليه .

بالنسبة لمطور الإنترنت تصبح عناصر التحكم ActiveX هامة حيث يدعم متصفح ويب مثل : Internet Explorer تلك العناصر ، وعلى ذلك فإذا تضمنت صفحة ويب عنصر تحكم ActiveX فسوف يتمكن المستخدم من التفاعل معه .

تعمل عناصر تحكم ActiveX مثل الكائنات objects فهى تدعم الخصائص والصفات التى تتحكم بها فى البرنامج .

أكثر استخدام لعناصر ActiveX بالنسبة لمطورى صفحات ويب هو تحكم الإنترنت بلغة فيجوال بيزك Visual Basic ، ولإضافة مثل هذا التحكم إلى تطبيق Visual basic يحتاج المبرمج إلى إسقاط عنصر التحكم داخل إطار التطبيق باستخدام معالج

تطبيقات VB .

لغات النصوص البرمجية المكتوبة فى تطبيقات الإنترنت

لا يدعم متصفح ويب حاليا فقط صفحات HTML بل يدعم أيضا لغات وتطبيقات أخرى ، فالمتصفح أكثر من مجرد مكان لحفظ ActiveX ، ويدعم برامج جافا Java ولغات نصوص مكتوبة مثل جافا سكربت JavaScript وفيجوال بيزك سكربت VBScript .

تتيح تلك اللغات الخاصة دمج كود للتحكم بداخل كود HTML ليقوم بالتفاعل والاستجابة مع الكائنات الأخرى فى صفحة ويب ، وتتشابه لغة JavaScript مع لغة Java كذلك تتشابه لغة VBScript مع لغة Visual Basic ، ويدمج المبرمج نص VBScript مباشرة بداخل كود HTML ، وتعطى لغات النصوص المكتوبة Script قدرة أكبر على تحليل الشفرة المختفية وراء صفحات ويب .

لغة جافا Java

شهدت صفحات ويب تقدما بظهور لغة جافا Java التى صممت أساسا للتطبيقات المدمجة مثل : الوحدات النمطية الخاصة بالكمبيوتر داخل الآلات ثم أصبحت لغة قياسية مستخدمة لإنشاء صفحات ويب وتحويلها إلى وسط متفاعل متغير .

تحتوى لغة Java على العديد من الوحدات النمطية السابقة التجهيز ، وهى لغة موجهة للكائنات OOP مماثلة للغة C++ فى أوجه كثيرة .

للعمل فى مجال برمجة تطبيقات ويب يجب تعلم لغة Java السهلة فهى لغة صغيرة ، وتكمن قوتها فى صغرها حيث إن البرامج الصغيرة يمكن تحميلها سريعا من خلال صفحات HTML لتمثل صفحات ويب المتفاعلة مع مستخدمى شبكة الإنترنت .

وبالرغم من طبيعة النصوص المتشعبة Hypertext لشبكة ويب فإن كل ما تقوم به النصوص هو التنقل من صفحة لأخرى ، وكان عرض الصور والنصوص فى الإنترنت فريدا لكنه سرعان ما فقد بريقه مع الوسائط المتعددة فمستخدمو الشبكة فى حاجة إلى التفاعل لجذب انتباههم بأكثر من شاشات رسومية .

غيرت جافا Java التى ابتكرتها شركة صن Sun Microsystems فى منتصف

التسعينيات طريقة عمل مواقع ويب ، فلغة Java لغة برمجة لها سمات مماثلة للغة C++ فبدلا من استخدام متصفحات ويب لعرض البيانات تقوم المتصفحات بتحميل برامج مكتوبة بلغة جافا Java ، ثم يتم تشغيل هذه البرامج على جهاز المستخدم بدلا من تشغيلها عن بعد على الجهاز الخادم ، كما يمكن أيضا التفاعل معها .

ليست كل المؤثرات الصوتية والحركية على ويب منتجات لغة جافا Java ، لكن لغة جافا Java ارتقت بالتفاعل مع الحاسب إلى ما هو أبعد من ذلك .

يمكن أن تكتب نوعين من برامج Java :

* بريمجات جافا Java Applets وهى برامج صغيرة تنتقل مع لغة HTML ويتم تشغيلها على الحاسب الخاص بمستخدم ويب .

* تطبيقات جافا Java وهى برامج قائمة بذاتها لا تحتاج إلى متصفح ويب أو لغة HTML لتشغيلها .

الهدف الأساسى من استخدام لغة جافا Java هو وضع برامج ذاتية التشغيل على صفحات ويب حتى يتمكن المستخدمون من الحصول على مزيد من التفاعل على مواقع ويب .

عندما تعمل بلغة جافا Java ستسمع عن المحتوى ذاتى التشغيل الذى يشكل حجر الزاوية فى لغة جافا ، وتحتوى صفحة ويب على محتوى ذاتى التشغيل عبر أوامر لغة HTML فى صورة بريمج Java applet فمحتوى أى صفحة ويب يعد ذاتى التشغيل على الجهاز الخاص بالمستخدم .

عندما يقوم المستخدم بإدخال العنوان URL لصفحة (أو نقر رابطة تشعبية تنقله إلى صفحة) يقوم متصفح ويب بتحميل صفحة HTML وإعادة تنسيق نص الصفحة وفقا لأوامر لغة HTML ، وأيضا يقوم بعرض أى صورة ، وتحميل المحتوى ذاتى التشغيل حتى يتم تشغيله ، ويتم تنفيذ المحتوى ذاتى التشغيل مباشرة أو بعد وقوع حدث سبق تحديده مثل نقر الفأرة على منطقة معينة فى صفحة ويب .

إذا كان على المستخدم أن يتفاعل مع الموقع عن بعد فى جلسة سؤال وجواب مثل ملء طلب شراء كان على المستخدم ملء الطلب بكامله قبل التحقق من وجود أخطاء ، إذن فأى تفاعل مع الموقع كان لابد وأن يبدأ من المستخدم ثم يكون رد الفعل

المحكوم بسرعة الموقع ومعدل التدفق على الإنترنت .

عند زيارة موقع ويب ينشط برنامج applet تلقائيا بالإضافة إلى ذلك يقوم البرنامج بتحميل نفسه على الحاسب الشخصى ويتم تشغيله بمجرد أن تقوم بالحدث الذى يؤدى إلى تشغيله ثم يتم مسحه عند مغادرة الموقع دون أن يشغل مساحة دائمة على القرص الصلب .

عندما يكتب المبرمج صفحة ويب قائمة على لغة Java يريد أن يعمل البرنامج عن بعد على جهاز المستخدم بغض النظر عن نوع الجهاز الذى يستخدمه .

بينما تقوم معظم مترجمات لغات البرمجة بتحويل برامجها إلى لغة تشغيل تعتمد إلى حد كبير على الجهاز الذى تمت عليه الترجمة ، فإن لغة Java لا تفعل هذا ، بل تقوم بترجمة برامجها إلى نوع خاص من الوحدات النمطية يسمى bytecode لا يعتمد على الجهاز الذى تمت عليه الترجمة وفيما بعد يستطيع أى جهاز أن يقوم بتشغيل الوحدة النمطية عن طريق متصفح ويب المساند للغة Java .

عند كتابة تطبيقات الإنترنت يحتل التأمين جزءا كبيرا من اهتمام المبرمج ، فبرامج جافا Java applets قد تؤدى إلى مشكلات تأمين إذا لم يتم اتخاذ الاحتياطات اللازمة ، حيث لا يمكن الجزم ما إذا كان يتم تشغيل برنامج applet ، أو ما الذى يفعله البرنامج فى القرص الصلب أو ذاكرة الجهاز ، ولحسن الحظ طورت شركة صن لغة جافا منذ البداية لتكون لغة برمجة قائمة على شبكات الاتصال ، وبناء على ذلك يعد التأمين مبدأ متأصلا فى اللغة للمبرمج ولمستخدم الإنترنت .

لغة فيجوال جافا Visual J++

توجد العديد من الإصدارات للغة Java أشهرها هى لغة Visual J++ من إنتاج مايكروسوفت Microsoft وتكمن شهرتها فى أوجه الشبه بينها وبين لغة Visual Basic وتساند لغة Java بالكامل ، وهى لغة برمجة لإنشاء برامج applet مساندة للغة Java أو تطبيقات Java قائمة بذاتها ، وتحتوى على محرر ومصحح أخطاء debugger ومستندات مساعدة بالإضافة إلى مترجم Java compiler لترجمة برنامج Java إلى شفرة موحدة bytecode .