البرمجة بلغه C++

تعتبر لغة c++ من أشهر اللغات التي تتمتع بطابع القوة والمرونة لإنتاج أسرع البرامج وأفضلها أداءاً .وعلى الرغم من وجود العديد من لغات البرمجة الأخرى إلا انها تفتقر شمولية لغة++C وقوتها فاللغة تتميز بقابليتها على معالجة التطبيقات الكبيرة والمعقدة، والقوة في صيانة البرامج المكتوبة بها مما يوفر وقتاً في تصميم البرامج وتطويرها .

كيفية كتابة برنامج ب C++

سنبدأ بكتابة برنامج يعرض نصاً على الشاشة:-

```
//Program 1-1:
//This program will display a message on the screen.
#include<iostream.h>
main ( )
{
      cout <<''welcome to C++ !\n'';
return 0;
}
```

الخرج من البرنامج:

```
welcome to C++ !
```

التعليقات: Comments

```
// Program 1-1:
//This program will display a message on the screen.
```

يبدأ هذا السطر من البرنامج بالشرطة المزدوجة (//) الدالة على أن بقية السطر عبارة عن تعليق (comment)، تضاف التعليقات إلى البرامج لتساعد المبرمج أو أي شخص آخر قد يحتاج إلى قراءة البرنامج على فهم ما الذي يفعله البرنامج، لذا من المستحسن أن يبدأ كل برنامج في لغة C++ بتعليق يوضح الغرض الذي من أجله كتب البرنامج.

تستخدم الشرطة المزدوجة (//) إذا كان التعليق يمتد لسطر واحد فقـــط single-line comment.

هنالك نوع آخر من التعليقات يتيح لنا كتابة تعليقات تمتد إلى عدة أسطر multi-line comments ، نستطيع كتابة التعليق السابق على الصورة:

```
/* Program 1-1:
 This program will display a message on the screen
*/
```

يبدأ الرمز /* التعليق وينهيه الرمز */ . نجد أن نهاية السطر لا تعني انتهاء التعليق لذا يمكننا كتابة ما نشاء من أسطر التعليقات قبل الانتهاء بالرمز */.

مرشدات المهيئ (Preprocessor Directive):-

#include<iostream.h>

يسمى هذا بمرشد المهيئ Preprocessor directive، وهو عبـــارة عـــن تعليمة للمصرف أن يدرج كل النص الموجود في الملف iostream.h في البرنامج، وهـــو ملف يجب تضمينه مع أي برنامج يحتوى على عبارات تطبع بيانات على الشاشة أو تستقبل بيانات من لوحة المفاتيح.

يسمى iostream ملف ترويسة (header file)، وهنالك الكثير من ملفات الترويسة الأخرى، فمثلاً إذا كنا نستعمل في برنامجنا دالات رياضية كـــ ( )sin و( )cos نحتاج إلى شمل ملف ترويسة يدعى math.h، وإذا كنا نتعامل مـــع سلاســـل الأحـــرف سنحتاج للملف string.h. وعموماً هنالك عدد كبير من ملفات الترويسات التي يجـــب تضمينها على حسب طبيعة البرنامج ، تعتبر ملفات الترويسات جزء مهم من برامج لغـــة C++ وسنحتاج إلى شمل الملف iostream.h لتشغيل أي برنامج يقوم بعمليات إدخـــال وإخراج.

الدالة main :-

main( )

يبدأ تشغيل أي برنامج C++ من دالة تدعي ( )main، وهي دالة مستقلة ينقـــل نظام التشغيل التحكم إليها. وهي جزء أساسي في برنامج C++.

الأقواس بعد main تشير إلى أن main هي عبارة عن دالة. قد يحتوى برنامج C++ على أكثر من دالة إحداهما بالضرورة هي main. يحتوى البرنامج السابق على دالـــة واحدة.

يبدأ تنفيذ البرنامج من الدالة main حتى لو لم تكن هي الأولى في سياق البرنامج.

يتم حصر جسم الدالة main بأقواس حاصرة { } .

الخرج إلى الشاشة:-

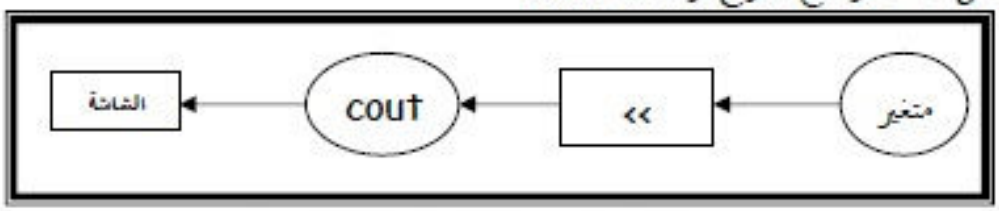
```
cout<<'' welcome to C++ !\n '';
```

هذه العبارة (statement) تجبر الحاسوب أن يظهر على الشاشة النص المحصور بين علامتي الاقتباس '' ''. ويسمى هذا النص ثابت سلسلي.

يجب أن تنتهي كل عبارة في برنامج C++ بفاصلة منقوطة ; (semi colon).

الاسم cout والذي يلفظ كـــ C out يمثل كائن في C++ مقترن مع الشاشة والعامل >> والذي يسمى بعامل الوضع Put to operator يجبر على إرسال الأشياء التي على يمينه إلى أي شئ يظهر على يساره.

الشكل ١-١ يوضح الخرج بواسطة cout.



مثال: شكل (١-١) الخرج بواسطة cout

```
//Program 1-2: Output
#include <iostream.h>
main ( )

{
      cout  << 7 << " is an integer.\n";
      cout << 'a' << "is a character.\n";
}
```

الخرج من البرنامج:

```
7  is an integer.
a  is a character
```

من خرج البرنامج يتضح لنا الآتي:

١- يتم حصر النص المطلوب ظهوره على الشاشة بين علامتي اقتباس" is an integer ".

٢- تتم كتابة الثوابت الرقمية بدون علامتي اقتباس 7 <<.

٣- يتم حصر حرف واحد مطلوب ظهوره على الشاشة بعلامة اقتباس فردية 'a'<<.

تقوم بعض اللغات كـ Basic مثلاً بالانتقال إلى سطر جديد تلقائياً في نهاية كل عبارة خرج ، لكن C++ لا تفعل ذلك كما أن العبارات المختلفة والموضوعة في أسطر مختلفة لا تؤدي إلى ذلك .

لا ينشئ الكائن cout أسطراً جديدة تلقائياً، والمخرجات في البرنامج التالي توضح

ذلك:-

```
//Program 1-3:This program displays output on the screen
#include<iostream.h>
main ( )
{
  cout<<10;
  cout<<20<<30;
  return 0;
}
```

تظهر الخرج:-

```
102030
```

حيث يلتصق كل الخرج ببعضه البعض ، لذا من الجيد أن يكون لدينا طرق في C++ للتحكم بطريقة تنسيق الخرج والتي منها تتابعات الهروب(Escape Sequences).

تتابعات الهروب (Escape Sequences):

نلاحظ أنه لم تتم طباعة \n على الشاشة ، \ تسمى الشرطة الخلفيـــة (Back slash) أو حرف هروب (Escape character) وتسمى هي والحرف الذي يليها تتابع هروب. تتابع الهروب \n يعني الانتقال إلى سطر جديد حيث يجبر المؤشر على الانتقال إلى بداية السطر التالي ، الآن إليك بعض تتابعات الهروب الشائعة:-

تتابع الهروب	الوصف
\n	سطر جديد.
\t	مسافة أفقية.
\b	حرف التراجع back space.
\\	لطباعة شرطة خلفية.
\r	حرف الإرجاع، يجبر المؤشر على الانتقال إلى بداية هذا السطر.
\''	لطباعة علامة اقتباس

العبـــــارة return 0 :-

تكتب العبارة return 0; في نهاية الدالة ( )main. القيمة 0 تشـير إلى أن البرنامج انتهى نهاية صحيحة وسيبدو لنا سبب تضمين هذه العبارة واضـحا عنـدما نتعرف على الدوال في C++ بالتفصيل.

مثال آخر لبرنامج C++ :-

إليك الآن مثالاً لبرنامج يستقبل رقمين من المستخدم ويجمعهما ويعرض ناتج الجمع:-

```
// Program 1-4: Addition program
#include<iostream.h>
#include<conio.h>
main ( )  {
      int integer1, integer2, sum;
      cout <<"Enter first integer\n";
      cin >> integer1;
      cout <<"Enter second integer\n";
      cin >> integer2;
      sum= integer1+integer2;
      cout <<"sum="<<sum<<endl;

return 0;
  }
```

أنواع البيانات الأساسية في لغة C++

هنالك سبعة أنواع بيانات أساسية في C++ ، واحد منها يمثل الأحرف وثلاثة تمثل أرقاماً كاملة (أعداد صحيحة) وثلاثة تمثل أرقاماً حقيقية. الجدول الآتي يلخـــص هـــذه الأنواع.

اسم النوع	يستعمل لتخزين	أمثلة عن القيم المخزنة
char	أحرف	''a''
short	أرقام صحيحة قصيرة	222
int	أرقام صحيحة عادية الحجم	153,406
long	أرقام صحيحة طويلة	123,456,789
float	أرقام حقيقية قصيرة	3,7
double	أرقام حقيقية مزدوجة	7,533,039,395
long double	أرقام حقيقية ضخمة	9,176,321,236,01202,6

١/ الأحرف char :-

يتم تخزين الأحرف في متغيرات من النوع char العبارة:-

```
char ch;
```

تنشئ مساحة من الذاكرة لحرف وتسميه ch. لتخزين حرف ما في هذا المتغير نكتب

```
ch='z'
```

ودائماً تكون الأحرف الثابتة كـــ 'a' و'b' محصورة بعلامة اقتباس فردية.

يمكن استعمال المتغيرات من النوع char لتخزين أرقام كاملة بدلاً من أحرف ، فمثلاً يمكننا كتابة:-

```
ch=2;
```

لكن نطاق القيم الرقمية التي يمكن تخزينها في النوع char يتراوح بين 128- إلى 127 لذا فإن هذه الطريقة تعمل مع الأرقام الصغيرة فقط.

٢/ الأعداد الصحيحة:

تمثل الأعداد الصحيحة أرقاماً كاملة أي قيم يمكن تعدادها ، كعدد أشخاص أو أيام أو عدد صفحات مثلاً ، ولا يمكن أن تكون الأعداد الصحيحة أرقاماً ذات نقطـة عشـرية ولكنها يمكن أن تكون سالبة.

هنالك ثلاثة أنواع من الأعداد الصحيحة في C++: short قصير، int عدد صحيح، long طويل وهي تحتل مساحات مختلفة في الذاكرة. الجدول التالي يبين هذه الأنواع والمساحة التي تأخذها في الذاكرة ونطاق الأرقام التي يمكن أن تأخذها:

اسم النوع	الحجم	النطاق
char	1byte	-128 إلى 127
short	2byte	-32,768 إلى 32,767
int	مثل short في أنظمة 16bit ومثل long في أنظمة 32bit	
long	4byte	-2,147,483,648 إلى 2,147,483,647