الدرس الثاني : الخطوات الأولى في عالم السي شارب .

أهلا و سهلا بك من جديد في عالم السي شارب ، كنت قد وعدتك بأن أشرح لك معنى ا لشفرة التي كنت قد كتبتها لك فهيا بنا .

الخطوة الأولى :

دعني أذكرك بتلك الشفرة :

```
using System;
class FirstPor
{
      static void Main ()
      {
            Console.WriteLine ("Hello, World ");
      }
}
```

قبل أن أبدأ في شرح مفردات هذه الشفرة ، أود أن أخبرك شيئا عن لغة سي شارب ؛ ألا وهو أنها لغة كائنية التوجيه **Object Oriented Programming** ، و معنى هذا أن كل شيء في هذه اللغة هو كائن ؛ فالقائمة المنسدلة كائن ، و مربع النص كائن ، و مربع الرسالة كائن ، و برنامجك السابق كائن ، و كل كائن له خصائص و أفعال يستطيع فعلها ، و بما أن لغة سي شارب كائنية التوجيه فإن أي برنامج يجب أن يكون متكون من كائن أو أكثر ، و يعرف الكائن – الذي يسمى في البرمجة بالخلية – بالكلمة المحجوزة class متبوعة باسم ذاك الكائن وهو هنا **FirstPor** – هنا أود أن ألفت انتباهك أني سوف أترجم كلمة Class بكلمة خلية ، و جمعها خلايا – ، و كل خلية تبدأ بقوس معقوف { و تنتهي بقوس معقوف آخر } و ما بينهما هو محتويات تلك ا لخلية .

ستقول طيب ما فائدة السطر الأول ؟؟

سأقول إن لهذا السطر قصة طويلة ، ملخصها أن منصة الدوت نت – التي تبرمج عليها حاليا – منظمة في أقسام معينة ، و كل قسم متفرع إلى أقسام أخرى ، و حتى تستخدم أي خلية أو وسيلة من خلية معينة يجب عليك أن تكتب اسمها بالكامل فيمكن أن تكتب السطر السادس كالتالي :

```
System.Console.WriteLine ("Hello World ");
```

و تحذف السطر الأول ، و سوف يعمل مع البرنامج بشكل طبيعي ، فتصور الآن أنك سوف تكتب مائة سطر على هذا الشكل ، بدون شك سوف يصبح تكرار كلمة **System** بدون فائدة ، و مضيعة للوقت والجهد ؛ فاختصارا و تسهيلا علينا نحن معشر المبرمجين اخترعوا لنا كملة **using** و التي تجعل البرنامج

يعرّف أن جميع الخلايا التي سوف نستخدمها موجودة في اسم الفضاء الذي يلي كملة **using** ، و لها أيضا استعمالات أخرى سوف نذكرها إذا سمحت لنا الفرصة.

و الآن فلنتقل إلى السطر الثالث ، كما تعرف فإن كل برنامج تكتبه بالسي شارب يجب أن يحوي على الأقل على خلية واحدة ، و هي بدورها تحوي على بعض الخصائص و الأفعال التي تستطيع أن تقوم بها ، تسمى هذه الأفعال بالوسيلة **method** و جمعها وسائل .

و لكي يعمل برنامجك بشكل صحيح ، يجب أن تحوى إحدى الخلايا برنامج على وسيلة **Main** ، بحيث تكون نقطة بداية تنفيذ البرنامج .

و كل وسيلة يجب أن تحوى متغيراتها بين قوسين ( ) فإذا كانت لا تحوي على متغيرات فتترك فارغة – لا تقلق إذا لم تفهم هذه العبارة سوف نشرحها بالتفصيل فيما بعد بإذن الله ، و تعريف الوسيلة محصور بين قوسين معقوفين كالشكل التالي :

```
class  CalssName
{
      firstMethod  ( معاملات الوسيلة الأولى)
          {
               محتوى الوسيلة الأولى
          }
     SecondMethod (معاملات الوسيلة الثانية)
          {
               محتوى الوسيلة الثانية
          }
}  نهاية الخلية الأولى
```

أما الكلمة **void** فإنها تعني أن الوسيلة لا ترجع شيء ، أما **static** فلا تقلق منها الآن ، فقط ما عليك إلا أن تكتبها هكذا و سوف نشرحها فيما بعد ...

والآن وصلنا إلى قلب البرنامج ألا وهو السطر السادس ، حيث أنه هو المسئول عن اظهر جملة **" Hello, World "** ، كل الذي فعلناه هو أننا قمنا بمناداة الوسيلة WriteLine من الخلية Console ثم بعد ذلك قمنا بإعطائها العبارة التي سوف تظهرها على الشاشة ، مع تحديدها بعلامة الاقتباس **" "** حتى يعرف المترجم أننا نريد أن نضع جملة نصية و ليس شيئا أخر .

والآن جرب أن تكتب برنامج يقوم بإظهار اسمك وعنوانك و بريدك الإلكتروني ، مستخدما نفس الوسيلة ، بطبيعة الحال يمكنك أن تناديها أكثر من مرة ، و كل مرة تكتبها في سطر جديد .

هيا هيا تحرك إني انتظرك تنتهي من هذه المهمة ....

والآن بعد أن كتبت هذا البرنامج ، أظن أنك واجهت بعض المشاكل ، حيث أتوقع أن تصرخ في وجهي قائلا أني نفذت ما قلت لي بالحرف الواحد ولكن تظهر لي أخطاء كثيرة ، و قبل أن يتطور الموقف إلى بعض اللكمات ، سأقول لك أن هناك احتمالان لحدوث هذه الأخطاء ، أولا : أني تكون النسخة الفيجوال استديوا التي معك فيها أخطاء برمجية ، فيجب عليك في هذه الحالة أن تعلم شركة مايكروسوفت بهذا و تستبدلها بواحدة جديد خالية من الأخطاء ☺ ، و الاحتمال الثاني أن تكون قد ارتكبت بعض الأخطاء الإملائية عند كتابة شفرتك - رجعنا إلى الإملاء و أيام زمان - حيث أنك يجب عليك الالتزام ببعض الأشياء أثناء كتابة شفرة السي شارب و منها :

1- لغة السي شارب حساسة لحالة الحروف ، هذه الحساسية ورثتها من أمها السي ++ ، فـ System يختلف عن system ، و Main تختلف عن main و هكذا .

2- يجب أتنهي كل سطر من سطور شفرة السي شارب بالفاصلة المنقوطة ; ما عدا عند تعريف اسم الخلية أو الوسيلة .

3- هناك بعض الشروط لتسمية أسماء الخلايا منها : أن لا يكون اسم الخلية من الكلمات المحجوزة لسي شارب مثل class و for و هكذا ،و أن لا يحوي الاسم على فراغ مثل First Pro فهذا يعتبر اسم غير صحيح ،و أن لا تبدأ الأسماء بأرقام مثل 7amm ، و أن لا تحوي على رمز @ إلا في بداية الاسم.

الخطوة الثانية :

اكتب الشفرة التالية :

```
// البرنامج الثاني
using System;
class SecondPor
{
     static void Main ()
     {
     /* What is the differences between this methods ? */
          Console.Write ("Welcome to ");
          Console.WriteLine ("C# Programming");
     }
}
```

اعمل لها ترجمة و سترى المخرجات كالتالي :

```
Welcome to C# Programming
```

ستقول في نفسك أين ذهب الكلام الذي كتب باللون الأخضر و الذي يحوي على كلمات عربية و أخـرى إنجليزية ؟؟ ، سأقول لك إنها مجرد تعليقات ، تساعد على فهم البرنامج بالنسبة لك و لغيرك فاحرص كل الحرص على كتابتها ، فهي من العادات الجيدة لدى المبرمجين ، و طريقة عملها كما هو موضح في البرنـامج السابق إما بـ / / بحيث تكتب بعد هذه الإشارتين ما تشاء حتى ينتهـي الـسطر ، أو أن تكتـب تعليقـك محصورا بين هذه الإشارتين */ و / * .

و هناك نقطة أحب أن أوضحها في هذا الصدد ، ألا وهي أن السي شارب لا تهتم بالفارغ ما لم يكـن بـين علامتي التنصيص " " حيث يمكن أن تكتب البرنامج السابق كالتالي بدون أي مشاكل :

```
using System;class SecondPor{static void Main (){/* What
is the differences between this methods ? */Console.Write
("Welcome to ");Console.WriteLine ("C# Programming");}}
```

ولكن لي رجاء واحد ، وأتمنى أن لا تردني فيه ، أنك لا تفعل هذا في أي برنامج لأن حياتك سوف تتحول إلى الجحيم ، و عذاب لا يطاق ☺ .

و الآن نرجع إلى لب موضوع ما هو الفرق بين WriteLine و Write ، أتوقع منك أن تعرف الإجابة بكل سهولة ، الفرق هو يا عزيزي أن Write تضع مؤشر الكتابة حيث تنتهي الجملة التي سوف تظهرها ، أما WriteLine فتضعه في بداية سطر جديد ، أي كما في محرر النصوص عندما تنتهي الجملة نـضغط Enter لنبدأ سطر جديد ، فيكون تأثير WriteLine كطباعة الجملة أضف إليها Enter .

وسوف تدرك الفرق بنفسك حينما نعمل برنامج يطلـب مـن المـستخدم أن يـدخل بعـض المتغـيرات إلى برنامجنا .

و الآن اكتب ا لشفرة التالية :

```
// Third Program
using System;
class ThirdPor
{
      static void Main ()
      {
            Console.WriteLine ("Welcome\nto\nC#\nProgramming");
      }
}
```

و سوف تحصل على المخرجات التالية :

```
Welcome
to
C#
Programming
```

قد تتساءل لماذا ظهرت المخرجات هكذا ، و قد لا تتساءل ، صدقني أتمنى من عميق قلبي أن تتساءل ، و قد تكتشف بحسك البرمجي بأن هذا الرمز هو **\n** هو المسئول عن هذه ا لأفعال – يعمل كما أنه Enter- ، و سأقول لك ما شاء الله عليك أصبت كبد الحقيقة .

و أزيدك من الشعر بيت ، أن هناك العديد من هذه الرموز و التي تسمى سلاسل الهروب Escape Sequence ، لا تسألني الهروب من ماذا ، على العموم هي تستخدم لتنسيق المخرجات ، و الجدول التالي يوضح لك هذه الرموز :

الوصف	سلسلة الهروب
سطر جديد ، بحيث يوضع المؤشر في بداية السطر التالي	**\n**
عملة مسافة الجدولة الأفقية tab	**\t**
إعادة المؤشر إلى بداية السطر الحالي	**\r**
يستخدم لطباعة الشق المعكوس \ backslash	\\
يستخدم لطباعة علامة الاقتباس	**\"**

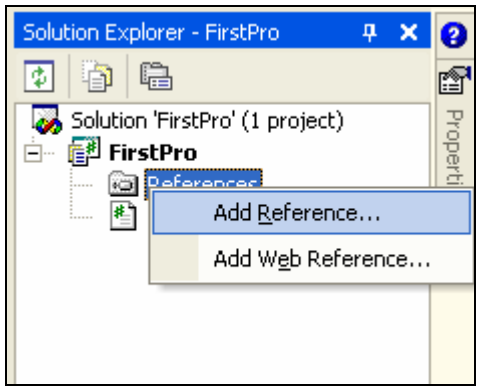
الخطوة الثالثة :

ما رأيك أن تكتب الشفرة التالية ، و ترى مخرجتها ؟؟

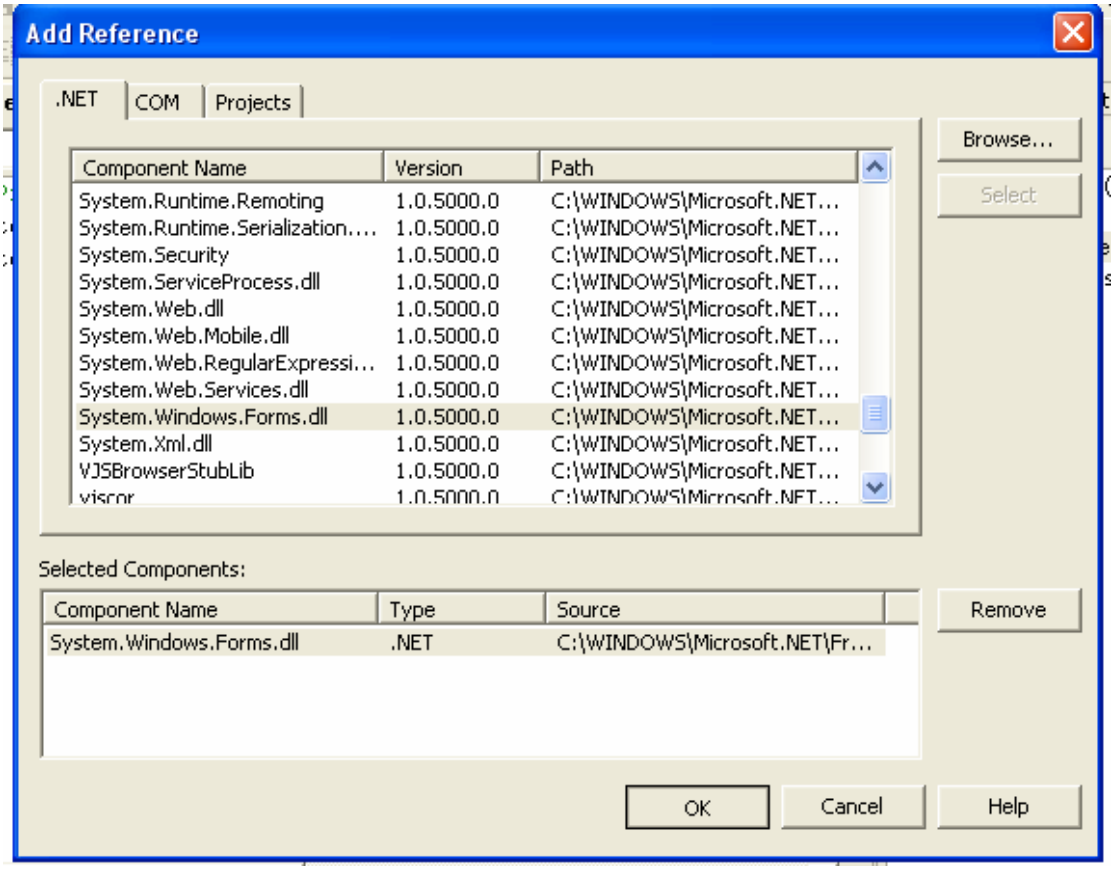
```
// Forth Program
using System;
using System.Windows.Forms;
class ForthPro
{
      static void Main ()
      {
      MessageBox.Show("Welcome\nto\nC#\nPorgramming");
      }
}
```

تمهل تمهل لا تستعجل ، إذا كنت سوف تستعمل الفيجول استديوا فإنه يجب عليك أن تضيف مرجع إلى مكتبة `System.Windows.Forms` في برنامجك ، لأنها ليست موجودة في مراجعك يمكنك التأكد

من ذلك بفتح ملف References من مستعرض الحلول Solution Explorer و لإضافة مرجع إليه اذهب إلى متصفح الحلول ، ومنه إلى مجلد المراجع ثم ا نقر باليمن و أختر Add References كما في الصورة التالية:



ثم سوف تظهر لك نافذة ، اختر التبويب **.NET**. ومنها اختر **System.Windows.Forms.dll** كما في الصورة التالية :



ثم Select ثم OK .

بعد ذلك ترجم البرنامج وسترى النتيجة التالية :

إنها مربع الرسالة من عالم الويندوز !!

أظن الكود السابق لا يحتاج شرح ، ما قمنا به هو مناداة الخلية MessageBox ثم استخدمنا الوسيلة Show لإظهار مربع الرسالة متضمنة الرسالة التي كتبنها كمتغير من متغيرات خلية الرسالة .

سوف تتمكن في القريب العاجل إن شاء الله أن تتحكم في كيفية ظهور مربع الرسالة بكامل الحرية بحيث تتمكن من التحكم في عدد الأزرار و شكل الأيقونات و عنوان الرسالة وغيرها من الأمور ولكن ليس الآن .

الخطوة الرابعة

ما رأيك أن نكتب برنامج يقوم بأخذ عددين من المستخدم و يجمعهما ثم يعرضهما ؟؟

هل أنت مستعد ؟؟ إذن فلنتوكل على الله :

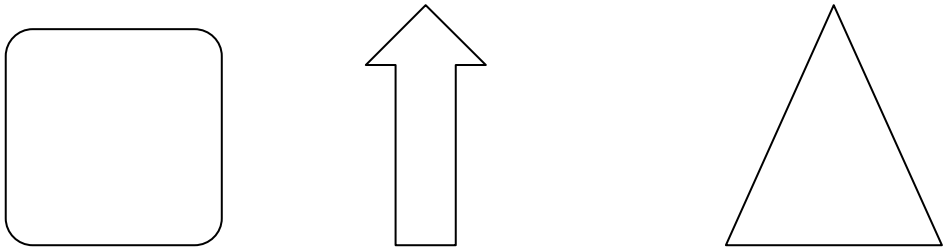
```
using System;
class Addition   // برنامج الجمع
{
      static void Main ()
      {
            string  FirstNumber, //متغير يحفظ العدد الأول من عن المستخدم
                    SecondNumber;//متغير يحفظ العدد الثاني من عن المستخدم
            int     x,          //متغير يحفظ العدد الأول كعدد صحيح
                    y,          //متغير يحفظ العدد الثاني كعدد صحيح
                    sum;        //متغير يحفظ الجمع
            // نطلب من المستخدم أن يدخل الرقم الأول
            Console.Write ("Please Enter the First Number: ");
            FirstNumber = Console.ReadLine();
            //نطلب من المستخدم أن يدخل الرقم الثاني
            Console.Write ("Please Enter the Second Number: ");
            SecondNumber = Console.ReadLine ();
            //نقوم بتحويل سلاسل الحرفية إلى أعدد صحيحة
            x =Int32.Parse (FirstNumber);
            y = Int32.Parse (SecondNumber);
            //نقوم بعملية الجمع هنا
            sum = x + y;
            //نعرض ناتج عملية الجمع
            Console.WriteLine ("The Sum = {0}",sum);
      }
}
```

سوف أقوم بشرح هذه الشفرة في الدرس القادم بإذن الله .

الواجب لهذا الدرس :

1 – قم بكتابة برنامج يقوم بعملية طرح لعددين ، يطلبهما من المستخدم .

2 – اكتب برنامج يقوم بعملية رسم لكل شكل من الأشكال التالية :



يمكنك رسمها باستخدام النقاط (.) أو النجوم (*) ، بطبيعة الحال استخدم Write و WriteLine ...

****
***

زايد السعيدي

منتدى الإبداع الإسلامي