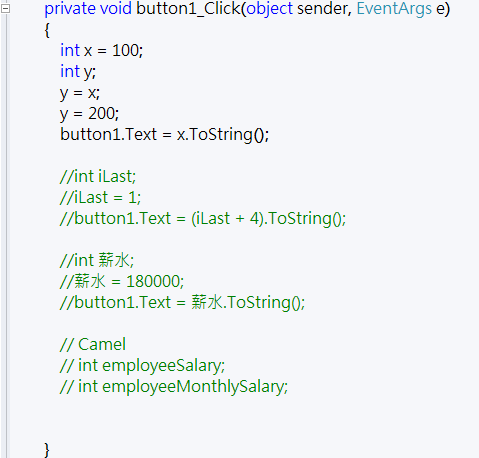
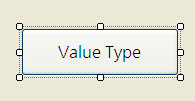
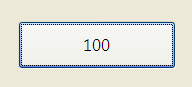
# 20150512 課程

課程重點:資料型別運用 範例檔20150512 DataType

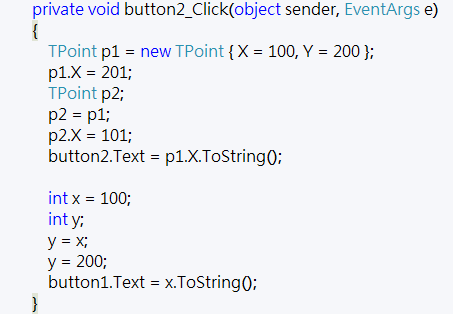
1. **Value Type (實值型別)︰常見的型別是 struct、int、char、double 等**
2. 新增 專案
3. 新增 button按鈕 修改button名稱 改為 “Value Type”
4. 點2下 進入 click事件 撰寫下列內容 如下圖



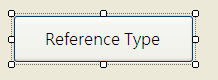
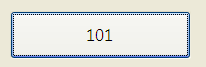
1. 結果



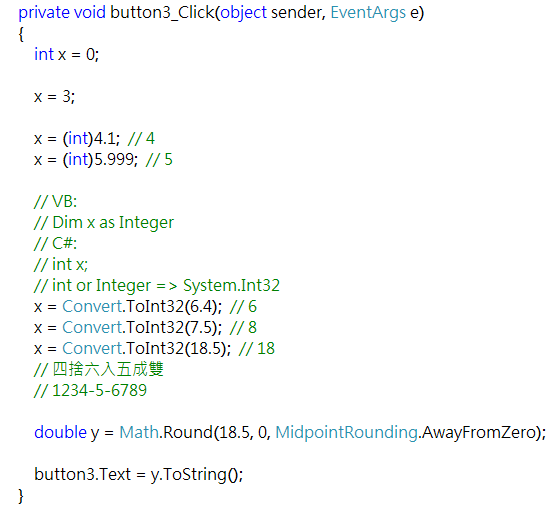
1. **Reference Type (參考型別)︰典型的例子就是使用 class 關鍵字定義的型別**
2. 新增 button按鈕 修改button名稱 改為 “Reference Type ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果



1. **Convert型別︰將基底資料型別轉換為其他基底資料型別。**
2. 新增 button按鈕 修改button名稱 改為 “Convert ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖

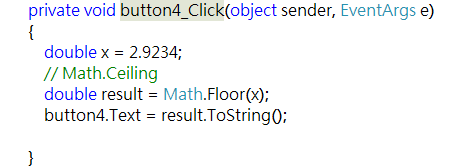


1. 結果

|  |  |
| --- | --- |
| [ToInt32(Byte)](https://msdn.microsoft.com/zh-tw/library/7hk2thbx(v=vs.110).aspx) | 將指定的 8 位元不帶正負號的整數值，轉換為相等的 32 位元帶正負號的整數。 |
| [ToInt32(Char)](https://msdn.microsoft.com/zh-tw/library/ww9t2871(v=vs.110).aspx) | 將指定的 Unicode 字元值轉換為相等的 32 位元帶正負號的整數。 |
| [ToInt32(Decimal)](https://msdn.microsoft.com/zh-tw/library/93kx4xke(v=vs.110).aspx) | 將指定之十進位數字的值，轉換為相等的 32 位元帶正負號的整數。 |
| [ToInt32(Double)](https://msdn.microsoft.com/zh-tw/library/ffdk7eyz(v=vs.110).aspx) | 將指定之雙精確度浮點數的值，轉換為相等的 32 位元帶正負號的整數。 |

1. **Math.Floor 方法︰傳回小於或等於指定數字的最大整數。**
2. 新增 button按鈕 修改button名稱 改為 “Floor ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖

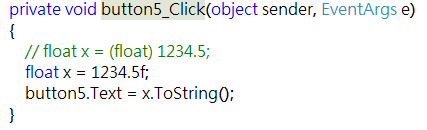


1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| **名稱** | **描述** |
| [Floor(Decimal)](https://msdn.microsoft.com/zh-tw/library/k7ycc7xh(v=vs.110).aspx) | 傳回小於或等於指定的十進位數字的最大整數。 |
| [Floor(Double)](https://msdn.microsoft.com/zh-tw/library/e0b5f0xb(v=vs.110).aspx) | 傳回小於或等於指定的雙精確度浮點數的最大整數。 |

1. **float方法︰float 關鍵字代表可儲存 32 位元浮點數值的簡單型別。。**
2. 新增 button按鈕 修改button名稱 改為 “數字f, 數字m ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖

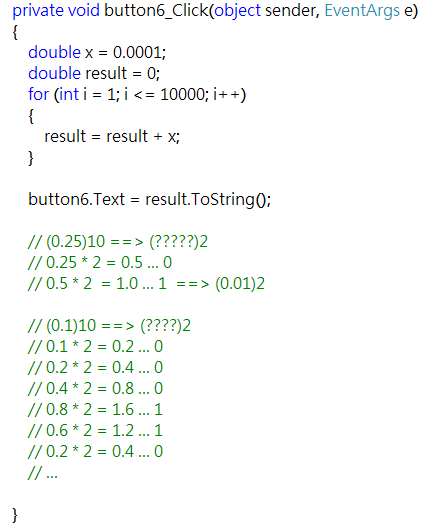


數字後面記得加上 “ F ”

1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

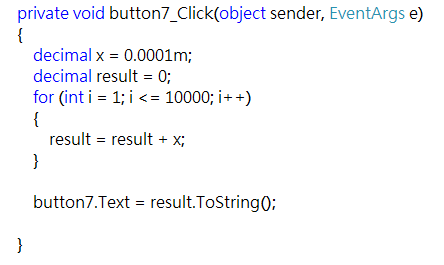
1. **浮點運算誤差方法︰**
2. 新增 button按鈕 修改button名稱 改為 “ 浮點運算誤差 ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

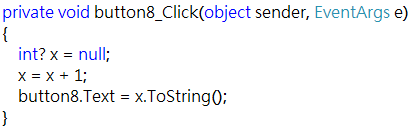
1. **nullable︰表示可以指派成 null 的實值型別。**
2. 新增 button按鈕 修改button名稱 改為 “ nullable ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

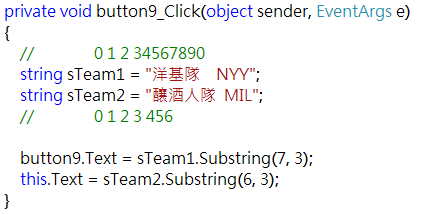
1. **Decimal︰表示十進位數字。**
2. 新增 button按鈕 修改button名稱 改為 “Decimal ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. **substring 函式:為第一個參數傳回從第二個參數所指定之位置開始的子字串，並傳回第三個參數中所指定的長度。**
2. 新增 button按鈕 修改button名稱 改為 “ Substring ”
3. 點2下 進入 click事件 撰寫下列內容 如下圖

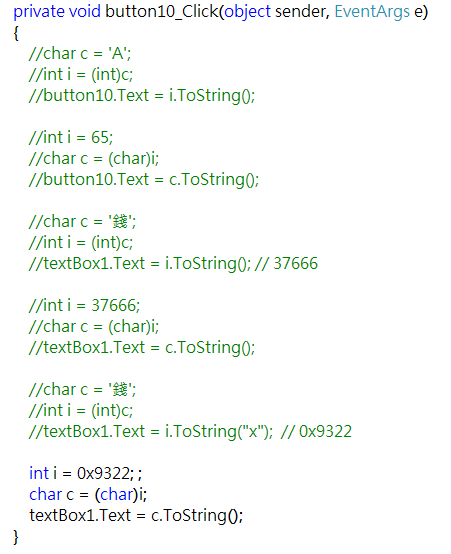


1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 參數 | 方法中定義的就叫做參數，例如 public int add(int a, int b)，此時的 int a 與 int b 就是參數，而引數就是方法裡面被用到的，如：  public int add(int a,int b)  {  return a + b;  }  裡面的 return a + b;裡面的a 就稱為引數　　return 裡面的b 也是引數 |
| 引數 | 一般我們稱為引數的是方法內部的那個變數，例如  substring(string a,int num, number?)  a num number?  這時候 都稱為要傳入的參數 |
| **參數和引數之間的差異 (Visual Basic)** <https://msdn.microsoft.com/zh-tw/library/9kewt1b3.aspx>  在大多數情況下，程序必須有一些可在其中呼叫程序之環境的相關資訊。 執行重複或共用工作的程序，每一次呼叫都使用不同的資訊。 該資訊是由呼叫程序時傳給它的變數、常數和運算式所組成。  **為了讓程序使用此資訊，程序將定義「參數」(Parameter)**，而**呼叫程式碼也會將「引數」(Argument) 傳給該參數**。 您可以將參數想像成停車位，將引數當做汽車。 就像不同的汽車能在不同的時間停入停車位，呼叫程式碼也可以在每次呼叫程序時，將不同的引數傳給同一個參數。 | |

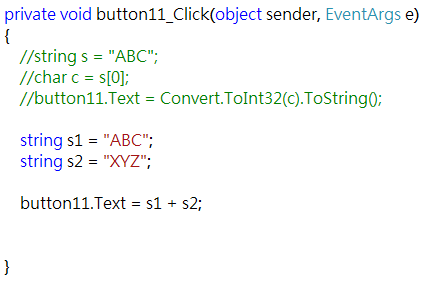
1. **char 函式: char 型別的常數可以寫成字元常值、十六進位逸出序列 (Escape Sequence) 或 Unicode 表示。**
2. 新增 button按鈕、textBox物件
3. 修改button名稱 改為 “ char ”
4. button按鈕點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

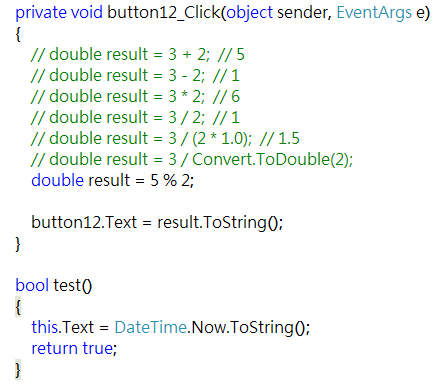
1. **Convert.ToInt32 方法: 使用指定之文化特性專屬格式資訊，將指定之物件的值轉換為 32 位元帶正負號的整數。**
2. 新增 button按鈕
3. 修改button名稱 改為 “Convert.ToInt32 ”
4. button按鈕點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

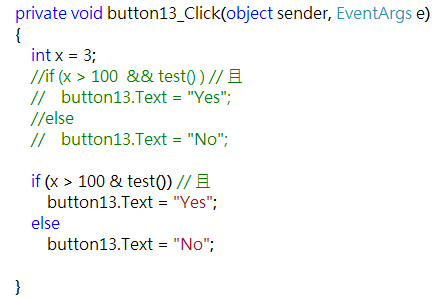
1. **bool 函式: 這個關鍵字是內建類型。 這個類型變數的值可以是 true 和 false。 條件運算式具有 bool 類型，因此具有 bool 類型的值。 例如，根據 i 的值而定，i!=0 現在會有 true 或 false。**
2. 新增 button按鈕
3. 修改button名稱 改為 “Convert.ToInt32 ”
4. button按鈕點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. **If-else 陳述式: 執行的陳述式會根據運算式的 Boolean 值的 if 陳述式識別。 在下列範例中，Boolean 變數 result 設為 true，然後以 if 陳述式進行檢查。 輸出為 The condition is true。**
2. 新增 button按鈕
3. 修改button名稱 改為 “ If-else ”
4. button按鈕點2下 進入 click事件 撰寫下列內容 如下圖



1. 結果

|  |  |  |
| --- | --- | --- |
|  |  |  |