

1 Escribir números con letras I

1.1 Sumario

En este documento se describe el funcionamiento de una DLL que he escrito y que permite escribir números con letras en un correcto español

1.2 Introducción

A la hora de escribir un número en letras (por ejemplo 28 = veintiocho) existen varios problemas derivados de la sintaxis del español que lo hacen relativamente complicado.

- Por ejemplo, veintiuno, veintiún plátano, veintiuna manzana.
- Por ejemplo cien (no un ciento) dos cientos
- Por ejemplo mil (no un mil) dos mil etc.
- Por ejemplo Un millón de plátanos un millón un plátano (obsérvese la preposición 'de')

1.3 Forma de leer los números

Para leer los números, el español usa el procedimiento denominado escala numérica larga que consiste en

- Numerar las cifras de derecha a izquierda, el orden de cada cifra es el lugar que ocupa en ese orden.
- Las cifras se agrupan de tres en tres de derecha a izquierda, cada uno de estos grupos de tres cifras se denomina clase, y se numeran también de derecha a izquierda.
- Agrupando las cifras de seis en seis de derecha a izquierda, o lo que es lo mismo, cada dos clases, se forman los periodos, que se numeran igualmente de derecha a izquierda.
- Hecha esta división, tenemos que el orden dentro de cada clase se denominan: unidad, decena y centena, la segunda clase dentro de cada periodo se denomina de millar, el segundo periodo son millones, el tercero billones, etc. En principio esta clasificación puede continuar indefinidamente.

Escribir números con letras [DLL] (I)

Por ejemplo, el número:

15936535897932384626433832795

Se clasificaría así:

15	936	535	897	932	384	626	433	832	795
10 ^a C	9 ^a C	8 ^a C	7 ^a C	6 ^a C	5 ^a C	4 ^a C	3 ^a C	2 ^a C	1 ^a C
5 ^o periodo		4 ^o periodo		3 ^e periodo		2 ^o periodo		1 ^e periodo	

Según esto y a la vista del esquema, el nombre de las cifras de izquierda a derecha sería:

- Unidad, decenas y centenas, para la primera clase del primer periodo.
- Unidad de millar, decenas de millar y centenas de millar, para la segunda clase del primer periodo.
- Unidad de millón, decena de millón, centena de millón, para la primera clase del segundo periodo.
- Unidad de millar de millón, decena de millar de millón, centena de millar de millón, para la segunda clase del segundo periodo.
- Este ciclo de seis cifras dividido en dos clases de tres cifras cada una que se denominan: Millón, Billón, Trillón, Cuatrillón, Quintillón, Sextillón, Septillón, Octillón, etc. puede nombrar cualquier cantidad por muchas cifras que pueda tener.

El número se lee:

- quince mil novecientos treinta y seis cuatrillones
- quinientos treinta y cinco mil ochocientos noventa y siete trillones
- novecientos treinta y dos mil trescientos ochenta y cuatro billones
- seiscientos veintiséis mil cuatrocientos treinta y tres millones
- ochocientos treinta y dos mil setecientos noventa y cinco

1.3.1 Otro Ejemplo

Dado el número:

31415926535897932384626433832795

Se separan sus cifras de tres en tres y se delimitan así:

31	5	415	926	4	535	897	3	932	384	2	626	433	1	832	795
12 ^a C	11 ^a C	10 ^a C	9 ^a C	8 ^a C	7 ^a C	6 ^a C	5 ^a C	4 ^a C	3 ^a C	2 ^a C	1 ^a C	0 ^a C	9 ^a C	8 ^a C	7 ^a C
6 ^o periodo		5 ^o periodo		4 ^o periodo		3 ^e periodo		2 ^o periodo		1 ^e periodo					

Leyéndose:

- Treinta y un quintillones
- cuatrocientos quince mil novecientos veintiséis cuatrillones
- quinientos treinta y cinco mil ochocientos noventa y siete trillones
- novecientos treinta y dos mil trescientos ochenta y cuatro billones
- seiscientos veintiséis mil cuatrocientos treinta y tres millones
- ochocientos treinta y dos mil setecientos noventa y cinco.

1.4 Clase [OperacionesEscribeNumeroConLetras]

Esta clase es la que encapsula las reglas necesarias para escribir un número en letras. Utiliza el concepto de Programación Descendente o Diseño TOP – DOWN

- [<http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r51758.PDF>]
- [http://es.wikipedia.org/wiki/Algoritmo_divide_y_vencer%C3%A1s]

La función que inicia el proceso de escritura de un número en letras es:

```
Private Shared Function auxEscribeUnNumero(  
    ByVal pCadenaNumerica As String,  
    ByVal pFormaDeEscrituraDelUno As FormatoEscrituraDelUno) As String
```

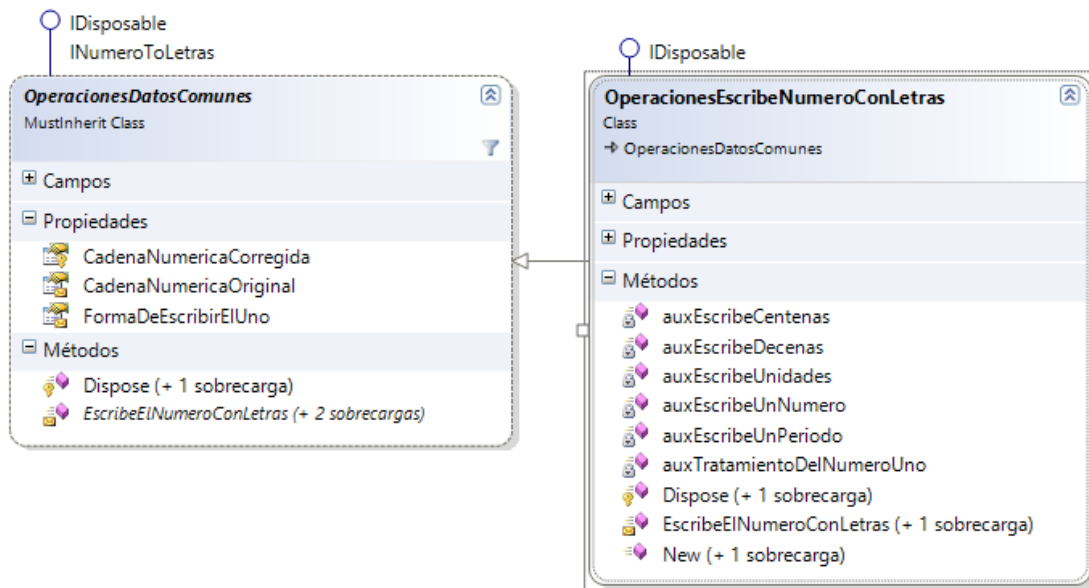
Esta función divide el número en [periodos]. Y llama a la función que se encarga de escribir el periodo. Cada periodo está compuesto de dos Clases (o centenas) de forma que cuando hay que escribir una centena se llama a la función que escribe las centenas. Esta llama a la función que escribe las decenas, y esta llama la función que escribe las unidades

La función tiene dos parámetros:

- La cantidad que se desea escribir en letras
- Un valor de la enumeración [FormatoEscrituraDelUno] que declara cual es el género del sustantivo que califica el número. Este valor se va transmitiendo a todas las funciones (descendentes) que se van llamando, de forma que en todo momento cada función sabe cómo hay que escribir el número.

Cada función se preocupa de escribir su parte del número con las peculiaridades propias. Por ejemplo, es mil, no un mil, por ejemplo, es cien ciento uno, etc.

A continuación se muestra el diagrama de las funciones y propiedades de la clase.

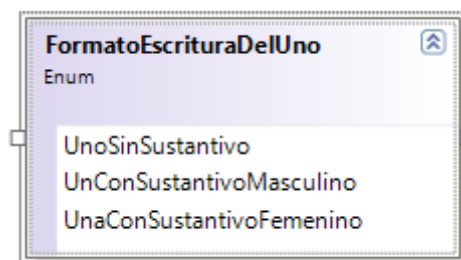


1.4.1 Escribiendo las unidades

Son diez dígitos con sus nombres propios, del cero al nueve. Cada dígito tiene un nombre propio e invariable, independiente del género del sustantivo que califica la cantidad. Por ejemplo tres peras, tres manzanas.

La excepción es el número uno (1) que puede tomar varias formas diferentes en función del sustantivo calificador que le acompañe. Por ejemplo [uno] sin calificador, [una] manzana si el calificador es femenino o [un] plátano, si el calificador es masculino.

Para tratar esta problemática, existe una enumeración que define (enumera) los tres casos posibles que se pueden dar al escribir un número terminado en uno (1)



1.4.2 Escribiendo las Decenas

Las decenas son los números comprendidos entre el 10 y el 99, y es en este bloque donde aparecen un montón de particularidades

Las decenas tienen un nombre único (diez, veinte, treinta... etc.)

Los números del 11 al 29 tienen un nombre propio, por ejemplo once, veintidós, veintinueve.

Los números del 31 hasta el 99 tienen un nombre compuestos (Por ejemplo treinta y dos, cuarenta y cinco)

Las decenas acabadas en uno cambian la escritura del [un] en función del sustantivo que acompaña al número. Cambian a [uno][una] solo si la palabra que sigue es un sustantivo masculino o un sustantivo femenino comenzando en «a» o en «ha» tónica. Ejemplo, veintiuno, veintiún plátano, veintiuna manzanas

1.4.3 Escribiendo las Centenas

En español, las [centenas](#) tienen varias irregularidades: La centena se expresa como [cien] si va sola y como [ciento] si va acompañada de decenas o unidades.

Ejemplo:

- cien manzanas cien, plátanos (no un cien)
- doscientas manzanas doscientos plátanos

Para expresar varias centenas, se usa el plural [cientos], uniéndose esta palabra al número que está multiplicando a [cien]. Observa que doscientos, trescientos, etc. (menos el cien) cambian según que la palabra siguiente sea masculina o femenina, por ejemplo doscientos zapatos, doscientas patatas

1.4.4 Escribiendo los Millares

Los [millares](#) son casi completamente regulares. La norma general es simplemente escribir los millares, seguidos de «mil», más el número de tres cifras que siga, excepto si la cifra de millares es la unidad (1). En este caso se omiten los millares, escribiendo sólo "mil" en lugar de "uno mil".

Aunque tradicionalmente se ha venido usando el punto (.) como separador de millares (p.ej.: 1.000), en la actualidad la RAE (Real Academia de la lengua Española) y el Diccionario panhispánico de dudas, el (DPD) dicen que al escribir números de más de cuatro cifras, se agruparán estas de tres en tres, empezando por la derecha, y separando los grupos por espacios en blanco: 8 327 451 (y no por puntos o comas, como, dependiendo de las zonas, se hacía hasta ahora: 8.327.451; 8.327,451. Y añade que, los números de cuatro cifras se escriben sin espacios de separación: 2458 y no 2 458. Pero sobre todo, en ningún caso deben repartirse en líneas diferentes las cifras que componen un número. O sea, que mientras las palabras sí se pueden partir siguiendo unas determinadas reglas, los números no se trocean y tienen que mantener todas sus cifras en la misma línea

Deberemos saber que nunca se escriben con puntos, comas ni blancos de separación, los números referidos a años, páginas, versos, portales de vías urbanas, códigos postales, apartados de correos, números de artículos legales, decretos o leyes. Año 2008, página 5123, código postal 50008, rúa del Percebe 1333.

1.4.5 Escribiendo los Millones

Los múltiplos enteros de un millón, utilizados como contadores de cosas, siempre van acompañados de la preposición «de»; por ejemplo: «tres millones de euros». Cuando se especifican cantidades menores al millón, dentro del mismo número, esta preposición no se utiliza, por ejemplo: «un millón setecientos mil euros ».

Los múltiplos de un millón siempre son masculinos: se dice «quinientos millones de personas» a pesar de que «persona» es femenino y «quinientos» pueda utilizarse en femenino. Sin embargo, se dice «quinientos millones quinientas mil personas».

1.4.6 Nombres de unidades Grandes

Los nombres de las unidades grandes, (de millón para arriba) tienen una secuencia de formación que se muestra en el listado siguiente:

Una observación

- $10(03)$ debe leerse 10 elevado a 3
- $10(24)$ debe leerse 10 elevado a 24
- $1\,000\,000(8)$ debe leerse un millón elevado a 8

Relación de algunos nombres de unidades Grandes Grandes

- $10(03) = 1\,000$ mil
- $10(06) - 1\,000\,000(1) = 1\,000\,000$ un millón
- $10(12) - 1\,000\,000(2) = 1\,000\,000\,000\,000$ un billón
- $10(18) - 1\,000\,000(3) = 1\,000\,000\,000\,000\,000\,000$ un trillón
- $10(24) - 1\,000\,000(4) = 1\,000\,000\,000\,000\,000\,000\,000\,000$ un cuatrillón
- $10(30) - 1\,000\,000(5) =$ un quintillón
- $10(36) - 1\,000\,000(6) =$ un sextillón
- $10(40) - 1\,000\,000(7) =$ un septillón
- $10(46) - 1\,000\,000(8) =$ un octillón
- $10(52) - 1\,000\,000(9) =$ un nonillón

- $10(60) - 1\,000\,000(10) =$ un decillón
- $10(66) - 1\,000\,000(11) =$ un un decillón

Escribir números con letras [DLL] (I)

- 10 (72) - 1 000 000(12) = un duo decillón
- 10 (78) - 1 000 000(13) = un tri decillón
- 10 (84) - 1 000 000(14) = un cuatri decillón
- 10 (90) - 1 000 000(15) = un quinti decillón
- 10 (96) - 1 000 000(16) = un sexti decillón
- 10 (102) - 1 000 000(17) = un septi decillón
- 10 (108) - 1 000 000(18) = un octo decillón
- 10 (114) - 1 000 000(19) = un nove decillón

- 10 (120) - 1 000 000(20) = un vigentillón '

- - 1 000 000(30) = un trigentillón
- - 1 000 000(100) = un centillón
- - 1 000 000(200) = un bicentillón
- - 1 000 000(300) = un tricentillón

1.4.7 La interfaz de la clase

Fíjate que todas las funciones anteriores están escritas según las técnicas del análisis descendente (Top-Down). [http://es.wikipedia.org/wiki/Algoritmo_divide_y_vencer%C3%A1s]

También puedes observar que todas las funciones son `private` por lo que nadie puede verlas fuera de la clase, es decir, hace falta un punto de entrada que permita llamar a la ejecución de estas funciones

```
''' <summary>
'''   Escribe en letras una numero ENTERO (sin parte decimal).
''' </summary>
''' <param name="pCadenaNumerica">
'''   La cadena que contiene el numero.
'''   El número debe ser entero. (Sin cifras decimales)
'''   Es responsabilidad del llamador asegurarse de que
'''   la cadena es un numero entero bien formado
''' </param>
''' <param name="pFormaDeEscrituraDelUno">
'''   <para>Enumeración <see cref="FormatoEscrituraDelUno">
'''       FormatoEscrituraDelUno</see>
'''   </para>
'''   <para>Indica la forma de escribir una terminación
'''       en [uno] por ejemplo [un], [uno], [una]</para>
''' </param>
''' <returns>
'''   <para>Una cadena que contiene escrito en letras el numero pasado</para>
'''   <para>si el numero es cero(0) devuelve una cadena con el valor [cero]</para>
'''   <para>si la cadena tiene el valor [Empty] se
'''       devuelve una cadena con el valor [Empty] </para>
''' </returns>
''' <remarks>
'''   Esta función debe ser llamada por la función que se encargue de
'''   escribir la parte entera y la parte decimal de cada número
''' </remarks>
```

Escribir números con letras [DLL] (I)

```
Public Function EscribeElNumeroConLetras(  
    ByVal pCadenaNumerica As String,  
    ByVal pFormaDeEscrituraDelUno As FormatoEscrituraDelUno) As String  
  
    ' corregir la cadena recibida  
    ' cargar la variable [_cadenaNumericaCorregida]  
    Me.CadenaNumerica = pCadenaNumerica  
  
    ' llamar a la función que hace el trabajo real de escribir en números  
    Return auxEscribeUnNumero(Me._cadenaNumericaCorregida, pFormaDeEscrituraDelUno)  
End Function
```

La propiedad [CadenaNumerica] es el punto de entrada de los datos, y se encarga de analizarlos y de cargar las variables internas de la clase, que de momento no interesa comentarlas

1.5 ¿Qué queda?

El código que se ha mostrado hasta ahora escribe en letras, únicamente, números enteros.

En un proceso real, por ejemplo una contabilidad, se trabaja con números decimales y con unidades.

Una observación importante: En español se escribe exactamente igual la parte entera de un número que la parte decimal del mismo, la única diferencia está en el sustantivo que califica cada cantidad, y, que, al poder ser masculino o femenino influye en la forma en la que se escribe el número

En resumen:

Falta por escribir una clase [Director] que reciba el número con decimales que se quiere escribir en letras, y el nombre de las unidades que califican el número tanto en la parte entera como en la decimal, después de procesarlas, devuelva la cadena con el número y las unidades correctamente escritos en letras y en español. Es decir, necesito escribir una clase que se encargará de:

- Recoger el número (con o sin decimales) que queremos escribir en letras
- Recoger la palabra que acompaña a la parte entera tanto en singular como en plural y también si es masculina o femenina, por ejemplo: [euro, euros, masculino]. Y los mismos datos para la palabra que acompaña a la parte decimal por ejemplo [céntimo, céntimos, masculino]
- Comprobar el número y asegurarse de que está bien formado según el estilo del idioma español es decir. El separador de millares es un punto (.) o un espacio (' '). El separador decimal es la coma (,) y si existe solo puede estar una vez, y, evidentemente, solo debe contener dígitos, (nada de letras)
- Separar la parte entera y la parte decimal del número
- Llamar a la clase [OperacionesEscribeNumeroConLetras], para convertir cada parte numérica (parte entera y parte decimal) en letras.

- Por último lo único que queda es juntar todas las cadenas para devolver el número escrito en letras acompañados de sus unidades correspondientes.

1.6 Referencia bibliográfica

- **Anexo: Nombres de los números en español**
- http://es.wikipedia.org/wiki/Nombres_de_los_n%C3%BAmeros_en_espa%C3%B1ol

- **Escalas numéricas larga y corta**
- http://es.wikipedia.org/wiki/Escalas_num%C3%A9ricas_larga_y_corta

- **Reglas para la escritura de los números cardinales**
- <http://www.saberespractico.com/estudios/cultura-general/reglas-escritura-numeros-cardinales/>

- Reglas ortográficas para la escritura de números
- <http://www.saberespractico.com/estudios/ortografia-numerica-escritura-de-numeros/>

- Hispanoteca - Como se escribe 'euro'
- <http://www.hispanoteca.eu/Foro-preguntas/ARCHIVO-Foro/Euro%20-%20euros.htm>

- Técnicas del análisis descendente (Top-Down).
[http://es.wikipedia.org/wiki/Algoritmo_divide_y_vencer%C3%A1s]
- [<http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r51758.PDF>]