

Re: coding an anagram function

Source:

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2006-10/msg03984.html>

- *From:* "NickHK" <TungCheWah@xxxxxxxxxxxxx>
 - *Date:* Fri, 20 Oct 2006 18:01:22 +0800
-

Any reason you cannot use the ASCII values of each character ? Or am I missing something ?

I assumed you want all spaces removed.

Not sure how you want to handle "a" and "A", so included a "CaseSensitive" argument you can toggle as required.

No checking that all values are actually in the alphabet. Also, if Unicode is used, you will have to amend.

I think it gives the correct results, with a quick bit of testing.

```
Public Function AreAnagrams(ByVal String1 As Variant, ByVal String2 As Variant, Optional CaseSensitive As Boolean = True) As Boolean
```

```
Dim Temp1() As Byte
```

```
Dim Temp2() As Byte
```

```
If CaseSensitive = False Then
```

```
'Change all to UCASE first
```

```
String1 = UCase(String1)
```

```
String2 = UCase(String2)
```

```
End If
```

```
'Remove any spaces
```

```
Temp1 = Replace(String1, " ", "")
```

```
Temp2 = Replace(String2, " ", "")
```

```
'See if they are the same length
```

```
If UBound(Temp1) <> UBound(Temp2) Then
```

```
AreAnagrams = False
```

```
Exit Function
```

```
End If
```

```
'Get the sum of the element values in each
```

```
'If not equal, cannot be anagrams
```

```
AreAnagrams = (SumElements(Temp1) = SumElements(Temp2))
```

```
End Function
```

```
Private Function SumElements(argArr() As Byte) As Long
```

Re: coding an anagram function

```
Dim i As Long
```

```
For i = LBound(argArr) To UBound(argArr)
```

```
SumElements = SumElements + argArr(i)
```

```
Next i
```

```
End Function
```

NickHK

"N Ramsay" <neil@xxxxxxxxxxxx> wrote in message

news:1161333528.328394.107300@xx

Hi,

I need to create a VBA function which compares two cells to see if the contents are anagrams of each other. Result of function would be true / false.

The cells will only contain letters, and no letter will appear more than 9 times. Each cell will never have any more than 40 characters in total. Spaces can be ignored.

The logic I was planning to use was to assign every letter of the alphabet a numeric value and then add up the numeric values of each string to give a numeric result.

For this to produce a unique result for any given string, i was planning to use values like the following:

a=1
b=1.1
c=1.01
d=1.001
e=1.0001
f=1.00001
g=10
h=10.1
i=10.01
j=10.001
k=10.0001
l=10.00001
m=100
n=100.1
o=100.01
p=100.001
q=100.0001
r=100.00001
s=1000
t=1000.1
u=1000.01

Re: coding an anagram function

v=1000.001
w=1000.0001
x=1000.00001
y=10000
z=10000.1

Given that no letter can appear more than 9 times, I believe this should return a unique result for every possible string of letters.

So, if the function compares two strings and gets the same addition based on the above rules, the strings must contain the same letters and are therefore anagrams of each other.

However, I have no idea how to code this in VBA.

Can anyone either suggest code for this, or another way of comparing two strings to see if they are anagrams of each other?

Many thanks in advance,

Neil.