

# Re: Count Specific Total Numbers from a Set

---

*Source:*

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2007-11/msg00289.html>

---

- *From:* "Dana DeLouis" <[ddelouis@xxxxxxxxxxxxxx](mailto:ddelouis@xxxxxxxxxxxxxx)>
  - *Date:* Fri, 2 Nov 2007 11:40:49 -0400
- 

Hi Paul. Here are a couple of ideas.

```
Total = Total + 1 ' <<<<<<<<< ADDED >>>>>>>>>
```

Notice that we are running the Total command above 13 million times. However, the Total is also the SUM of the 7 numbers in NPrime (0 to 6) I would skip that Total line, and just add the 7 values in the Solution array.

In the code below, notice that you are displaying the Total on each loop. You probably meant to do it only once at the end.

```
For R = 0 To 6  
Cells(R + 1, 1) = R  
Cells(R + 1, 2) = Format(NPrime(R), "#,0")  
Cells(R + 2, 2) = Format(Total, "#,0") ' <<<<<<<<< ADDED >>>>>>>>>  
Next R
```

Technique: Select the line of code with " For R = 0 To 6" in it from above, and hit the F9 key. This will place a Break-Point on this line. Now, run the code. The code will stop as it enters the loop. Now, hit the F8 key, and step thru the code. You will see the logic error now.

I just can't seem to be able to incorporate the Select Case construct into the Sub itself.

I factored out the Sub in case you wanted to run other tests. Suppose you wanted a count of the number of Even or Odd numbers, or some other idea. This way, it would be easy to adjust the code. We are only looping 49 times, so I don't think this is a real factor on the timing of the program. Anyway, here are some ideas:

```
Sub Prime()
```

## Re: Count Specific Total Numbers from a Set

```
Dim A As Long, B As Long, C As Long
Dim D As Long, E As Long, F As Long
Dim R As Long
Dim S As String
Dim Tme As Double
Dim Total As Long
```

```
' Are Prime ? (Yes = 1, No = 0)
Dim n(1 To 49) As Long
' Hold solution count
Dim NPrime(0 To 6) As Long
```

```
Tme = Timer
```

```
' Do Once Here
For R = 1 To 49
Select Case R
Case 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47
n(R) = 1
Case Else
n(R) = 0
End Select
Next R
```

```
For A = 1 To 44
For B = A + 1 To 45
For C = B + 1 To 46
For D = C + 1 To 47
For E = D + 1 To 48
For F = E + 1 To 49
R = n(A) + n(B) + n(C) + n(D) + n(E) + n(F)
NPrime(R) = NPrime(R) + 1
Next F
Next E
Next D
Next C
Next B
Next A
```

```
' Display Solution:
```

```
For R = 0 To 6
Cells(R + 1, 1) = R
Cells(R + 1, 2) = Format(NPrime(R), "#,0")
Total = Total + NPrime(R)
Next R
```

```
S = Format(Total, "#,0")
Cells(R + 1, 2) = S
```

```
// Stop timer before MsgBox
```

Re: Count Specific Total Numbers from a Set

Re: Count Specific Total Numbers from a Set

```
Debug.Print Timer - Tme
```

```
// Check
```

```
If Total = WorksheetFunction.Combin(49, 6) Then
```

```
MsgBox "Total is correct: " & S
```

```
Else
```

```
MsgBox "Error in Total: " & S
```

```
End If
```

```
End Sub
```

```
--
```

```
Dana DeLouis
```

```
<snip>
```

```
.
```