

Téma vyhledávání první prázdné buňky zdola ve sloupci je natolik frekventované, že jej znovu otevírám tímto článkem.

Máte ve sloupci výpis hodnot a další potřebujete přidat pod poslední vyplněnou buňku.



První prázdná
buňka zdola -
zadání

Chytřejší kolega vám pošle níže uvedený kus kódu.

1	Sub VyberPrvniPrazdnouBunkuZdola()
2	
3	Cells(65536, 1). End (xlUp).Offset(1, 0). Select
4	
5	End Sub

Procedura neřeší naše zadání přímo, staví se k problému z opačné strany. Od buňky ležící v průsečíku řádku 65536 a prvního sloupce postupuje směrem vzhůru (xlUp, běžně CTRL+šipka nahoru), dokud nenarazí na první vyplněnou buňku. Od ní se poté posune o jeden řádek dolů. Číslo 65536 by mělo vyjadřovat poslední řádek listu. To ovšem platilo v dobách Excelu 2003 a starších binárních souborech (xls). Dnešní listy sešitů (xlsx) mají řádků mnohem více (stačí podržet stisknuté tlačítko myši na místě, kde se potkává záhlaví řádků a sloupců a podívat se do Pole názvů) a je proto vhodné namísto této konstanty uvádět dynamicky Rows.Count. Další zpravidla neřešenou, ale podstatnou skutečností je to, že procedura chybně označí (jako první prázdnou) buňku z druhého řádku, pokud je sloupec celý prázdný (nemá hlavičku). Proč? Excel (VBA) má při použití .End(xlUp) nebo třeba při výběru využití oblasti listu (UsedRange) tendenci vždy vybrat alespoň jednu buňku listu (A1), a to i za cenu, že je prázdná, resp. do ní uživatel nikdy nezasáhl (nový list). Kód bych tedy upravil následovně:

```

1  Sub PrvniPrazdna()
2
3  Dim intSloupec As Integer
4  Dim rngTemp As Range
5  Dim strPrvniPrazdna As String
6
7  'zpracováváný sloupec
8  intSloupec = 1
9
10 'první neprázdná buňka zdola
11 'nefunkční pro celý prázdný sloupec
12 'strPrvniPrazdna = Cells(Rows.Count, intSloupec).End(xlUp).Offset(1, _
13     0).Address(0, 0)
14
15 'úprava
16 Set rngTemp = Cells(Rows.Count, intSloupec).End(xlUp)
17
18 'je buňka prázdná?
19 If IsEmpty(rngTemp) Then
20     'ano (prázdná buňka A1)
21     strPrvniPrazdna = rngTemp.Address(0, 0)
22 Else
23     'ne, posun o jednu níže
24     strPrvniPrazdna = rngTemp.Offset(1, 0).Address(0, 0)
25 End If
26
27 '.End(xlUp) zastaví:
28 'prázdný řetězec,
29 'prázdný řetězec převedený na hodnotu,
30 'hodnota skrytá formátem (;;;),
31 'nula skrývaná přes Možnosti / ...,
32 'apostrof
33
34 '.End(xlUp) nezastaví:
35 'skrytá hodnota (řádek) ať už přímo nebo filtrem
36
37 End Sub

```

Pamatujte, že technika .End(xlUp) ve svém hledání přeskakuje přímo či filtrem skryté buňky s hodnotou. Sloupec dat by přirozeně neměl obsahovat žádné kulišárny typ děr (prázdných buněk) mezi hodnotami atp. Pokud se na to můžete spolehnout, je možné si v úloze pomoci funkcemi listu POČET, případně POČET2 (zohledněte ve výsledku případnou hlavičku). Pro úplnost uvádím ještě jeden postup, který vychází z maticového vzorce listu

{=MAX(NE(JE.PRÁZDNÉ(Oblast))*ŘÁDEK(Oblast))}.

Jeho cílem je rovněž nalezení řádku poslední neprázdné buňky v oblasti. Aplikování pod VBA ve vlastní funkci (UDF) vypadá takto:

```
1 Sub Test()  
2  
3 Dim intSloupec As Integer  
4 Dim strPrvniPrazdna As String  
5  
6 'zpracovávaný sloupec  
7 intSloupec = 1  
8  
9 strPrvniPrazdna = epfSLOUPECPRVNIPRAZDNA(intSloupec)  
10  
11 End Sub  
12  
13 Public Function epfSLOUPECPRVNIPRAZDNA(ByVal Sloupec As Variant)  
14  
15 Dim strAdresa As String  
16  
17 'vynucený přepočít funkce  
18 Application.Volatile True  
19  
20 With ActiveSheet.UsedRange  
21 strAdresa = Replace(Columns(Sloupec).Address(0, 0), ":", _  
22 "1:") & .Cells(.Cells.Count).Row  
23 End With  
24  
25 epfSLOUPECPRVNIPRAZDNA = Cells(Evaluate("MAX(NOT(ISBLANK(" & _  
26 strAdresa & ")))*ROW(" & strAdresa & "))" + 1, _  
27 Columns(Sloupec).Column).Address(0, 0)  
28  
29 'prohledávání zastaví:  
30 'skrytá hodnota (řádek) ať už přímo nebo filtrem  
31 'prázdný řetězec,  
32 'prázdný řetězec převedený na hodnotu,  
33 'hodnota skrytá formátem (;;;),  
34 'nula skrývaná přes Možnosti / ...,  
35 'apostrof  
36  
37 End Function
```



První prázdná buňka zdola - vlastní funkce