

Pętla FOR...NEXT

Pętla ta (podobnie, jak w innych językach programowania) powtarza instrukcję (lub blok instrukcji) określoną ilość razy. Stosujemy ją zwykle wtedy, gdy wiadomo z góry ile razy ma się wykonać. Pętlę obsługuje zmienna, która za każdym razem zwiększa (lub zmniejsza) swoją wartość o 1 (można też zwiększać lub zmniejszać o inną wartość). Przyjęło się w wśród programistów, że ta zmienna sterująca oznaczana jest symbolem „i” (nie jest to oczywiście obowiązkowe) i dlatego czasami na pętli FOR...NEXT mówi się „iteracyjna”.

Pętla w pierwszej ramce wpisuje do 100 początkowych komórek kolumny pierwszej liczby od 1 do 100. W tym wypadku zrezygnowano z tradycyjnego „iterowania”, gdyż litera „w” wyraźniej symbolizuje wiersz komórki, do której wstawiamy.

Przykład w ramce środkowej wykonuje identyczną operację wpisywania 100 liczb, ale od tyłu: najpierw do komórki A100, A99, itd.

I ostatnia ramka – dokładnie takie same 100 liczb wpisane do tych samych komórek. Pętla kręci się tylko 50 razy, za każdym razem „i” zmienia się co 2 (1, 3, 5,...,99) i od razu kolejne liczby wpisywane są do dwóch komórek.

```
For w = 1 To 100
    Cells(w,1) = w
Next w
```

```
For w = 100 To 1 Step -1
    Cells(w, 1) = w
Next w
```

```
For i = 1 To 100 Step 2
    Cells(i, 1) = i
    Cells(i+1, 1) = Cells(i, 1)+1
Next i
```

Pętle zagnieżdżone

Przykład w ramce pokazuje pętle zagnieżdżone. Pętla zewnętrzna ze zmienną sterującą „w” wykonuje się 10 razy. Za każdym razem jednak wykonuje blok instrukcji, który zawiera pętlę ze zmienną sterującą „k” – pętla wewnętrzna. Instrukcja wewnątrz tych dwóch pętli wykona się łącznie 100 razy. Należy pamiętać by nie pomieszać początków i końców obu pętli – pętla zewnętrzna musi „obejmować” pętlę wewnętrzną. Być może teraz dopiero dostrzegasz wygodę stosowania wcięć.

```
For w = 1 To 10
    For k = 1 To 10
        Cells(w, k) = w * k
    Next k
Next w
```

Instrukcja FOR EACH...NEXT

Pewną odmianą pętli For... Next jest instrukcja For Each... Next, która służy do wykonywania operacji na takich samych obiektach arkusza, np. na komórkach. Praktycznym zastosowaniem może być np. przeszukiwanie komórek arkusza Excela w celu znalezienia określonej wartości. Przykład w ramce przeszukuje komórki z zakresu „A1:J10” – poprzednie makro wpisywało tam tabliczkę mnożenia). Jeżeli reszta z zawartości komórki jest równa zero (jest podzielna przez dwa), to komórka jest kolorowana na czerwono.

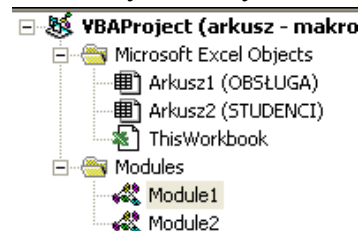
```
For Each kom In Range("A1:J10")
    If kom mod 2 = 0 Then
        kom.Interior.ColorIndex = 3
    End If
Next
```

Okienko powitalne podczas uruchamiania

Gdy uruchamiamy dowolny program komputerowy ustawiane są z reguły parametry początkowe, porządkowane paski narzędziowe, okienka, wczytywane dokumenty itp. W podobny sposób można przeprowadzić proces inicjalizacji w języku VB arkusza kalkulacyjnego. Przy okazji też warto zauważyć, że w ten sposób mogą uruchamiać się makrowirusy. Dlatego też w zabezpieczeniach makropoleczeń należy ustawić co najmniej średnie zabezpieczenie, aby za każdy razem, gdy uruchamiamy arkusz z makrami, można było o tym zdecydować.

Załóżmy, że po uruchomieniu arkusza będziemy chcieli wyświetlić okienko powitalne z komunikatem, zainicjować stałe globalne (będą obowiązywać w we wszystkich modułach) czy też wykonać dowolne inne operacje.

- przechodzimy do edytora makropoleczeń
- klikamy podwójnie w obiekt *ThisWorkbook* (lub prawym w ten obiekt i z menu wybieramy *View Code*) – przechodzimy do okienka edycji tego obiektu
- wstawiamy fragment kodu w ramce obok
- zapisujemy, zamykamy i ponownie otwieramy program
- po ponownym uruchomieniu, automatycznie jako pierwsze pojawi się okienko z komunikatem



```
Private Sub Workbook_Open()
    SkokDoStudentów
    MsgBox "Witam serdecznie"
End Sub
```

Oczywiście nic nie stoi na przeszkodzie, aby wykonać w tym okienku dowolne inne polecenie, w tym też makropolecenie przygotowane wcześniej w innych modułach – wystarczy wpisać nazwę tego makropolecenia. Przykładowo w ramce obok uruchomiono makro „SkokDoStudentów” znajdujące się w Module2.

Ćwiczenia

1. W tabeli STUDENCI tworzymy w kolumnie V (22) tekst powstały z połączenia nazwiska i imienia studenta rozdzielony spacją. Przycisk „PERSONALIA”.
2. Użytkownik będzie dokonywał potencjalnie niebezpiecznych operacji na arkuszu STUDENCI, dlatego też pisze makropolecenie, które będzie kopiować komórki arkusza z jednej kolumny do drugiej. W dwóch okienkach InputBox podajemy kolumny: z której - i kolumnę do której będziemy kopiować dane. Przycisk „KOPIUJ KOLUMNĘ”.
3. W tabeli STUDENCI, w kolumnach ZDAL i PRZYJETY występują wyrazy „tak” i „nie”. Wygodnie by było, gdyby cała tabela była ujednoczona i podobnie jak w kolumnie OLIMPIADA występowały litery „T” i „N”. Napisz makro, które dokona stosownych zmian. Przycisk „ZAMIEŃ TAK/NIE”.
4. Zauważono, że poprzednia wersja makra zamienia dobrze tylko jeden raz. za drugim razem pojawiają się same litery „N”. Procedura ulepszona sprawdza na wszelki wypadek tylko pierwsze litery wyrazów i niezależnie od tego czy wpisano wyrazy „Tak”, „TAK”, „tak”, czy same litery, czy w jakikolwiek inny sposób, zawsze pojawi się litera „T” lub „N”. Przycisk „ZAMIEŃ TAK/NIE ULEPSZONE”.
5. Miesięczne stypendium dla studenta obliczane jest według następującego algorytmu:
 - obliczamy iloraz punktów przez dochód i mnożony przez 10000
 - jeżeli student jest zamiejscowy, tzn. spoza Rzeszowa otrzymuje dodatek wysokości 100 zł
 - otrzymaną kwotę zaokrąglamy matematycznie do pełnych setek
 - stypendium otrzymują tylko studenci z oceną za egzaminy większą niż lub równą 4,0Wylicz stypendium dla wszystkich studentów w kolumnie W (23). Przycisk „STYPENDIUM”.
6. Za pomocą okienka InputBox wczytujemy numer studenta. Makro wyszukuje studenta o tym numerze i wybiera komórkę. Przycisk „SZUKAJ”
7. Pokoloruj kolejne wiersze tabeli naprzemiennie. Jeśli nie pamiętasz, jak pokolorować wiersz – uruchom nagrywanie. Przycisk „KOLORUJ”. Oryginalne makro z nagrywania w Module2.
8. Za pomocą okienka InputBox wczytujemy imię studenta. Makro koloruje tylko te wiersze, w których znajduje się wybrane imię. Przycisk „SZUKAJ i KOLORUJ”.
9. Napisz makro, które policzy ilu jest studentów w tabeli. Przeszukujemy wiersze aż do 10000 i zliczamy ilość niepustych komórek z numerem studenta. Przycisk „ILU”.
10. Napisz makro, które zaznaczy komórkę w pierwszym, pustym wierszu tabeli. Pusty wiersz to ten, w którym komórka z numerem jest pusta. Przycisk „PUSTY”.
11. Napisz makro, które wybierze wiersz studenta z najwyższym dochodem. Przycisk „MAX”.
12. Napisz makro, które w okienku MsgBox wypisze nazwiska i imiona studentów z najwyższym i najniższym dochodem. Przycisk „MIN-MAX”.
13. Zarejestruj makropolecenia, które będą sortowały tabelę STUDENCI według numerów i według nazwisk i imion. Przyciski „SORT NR” i „SORT NAZWISKO”. Procedury w Module3
14. Lista studentów w polu kombi (procedury w Module4)
Na arkuszu STUDENCI przygotuj pole kombi, które będzie zawierało posortowaną alfabetycznie listę nazwisk i imion. Lista ma być zawsze odświeżana po otwarciu pliku. Wszystkie czynności, które należy wykonać zostały już opisane w poprzednich ćwiczeniach.
 - przygotować makro, które tworzy listę z połączonymi nazwiskami i imionami np. w arkuszu obsługa
 - przygotować makro, które sortuje tę listę
 - utworzyć pole kombi i przypisać do niego zakres listy i komórkę na numer wybranego z listy elementu
 - przygotować makro, które pobiera z numer wybranego z listy nazwiska, szuka tego nazwiska i skacze do odpowiedniej komórki
 - przypisać to makro to pola kombi
 - w obiekcie ThisWorkbook utworzyć makro uruchamiające przygotowanie i sortowanie listy