

# MS Excel VBA za studente kemije

## - podsjetnik -

Ovaj podsjetnik sadrži kratka objašnjenja i pravilni način pisanja (sintaksu) za sve naredbe koje su obrađene tijekom održavanja *Računalnog praktikuma 2*.

### Option Explicit

Ova opcija nije nužna, ali kada je uključena u program, tada će program zahtijevati da se DEKLARIRA baš svaka varijabla koja se koristi. Deklaracija se vrši naredbom DIM (Dim dolazi od riječi "dimension"), u suprotnom se javlja greška. Za studente ova opcija je OBAVEZNA!

### Sub imeprograma ()

Početak svakog programa.

Ime ne bi trebalo sadržavati RAZMAKE, SPECIJALNE ZNAKOVE ili HRVATSKA SLOVA.

Velika i mala slova VBA ne razlikuje, pa je dovoljno pisati sve malim slovima, a ukoliko Excel sam postavi početno slovo Veliko, tada se radi o rezerviranoj riječi koja ne smije biti naziv programa. Ponekad se zbog iste greške javi poruka "Compile Error: Expected: Identifier".

### Worksheets ("Sheet1").Activate

(Umjesto "Sheet1" staviti točno ime radnog worksheeta)

Ova opcija izričito daje programu do znanja gdje se točno nalaze podaci sa kojima se radi. Opcija nije nužna, ali je korisna kada radimo sa nekoliko *worksheetsa* istovremeno.

Ako je ova linija izostavljena, program će raditi sa podacima iz worksheeta u kojemu je zadnje nešto napravljeno. Naredba ne funkcionira ako programi nisu napisani u **modulima**.

### Sheet1.Cells.Clear

Služi za brisanje prethodno upisanih sadržaja u čitavom worksheetu. Opcija nije nužna, ali je korisna.

### Dim k As Integer (ili String ili Double ...)

Obavezno dolazi prije prvog spominjanja varijable u programu. Najbolje odmah na početku.

Ovisi o očekivanim vrijednostima varijable (vidi tablicu "**vrste podataka u VBA**!")

U praksi se najčešće koriste navedene tri (alfa)numeričke deklaracije.

```
Const c As Double = 300000000
```

Na sličan način kao i varijable, deklariraju se i konstante.

Treba paziti da vrsta podatka odgovara vrijednosti konstante:

Ne može se pisati:

```
Const pi As Integer = 3.14     ✗
```

... jer će u tom slučaju program zaokružiti pi na cijeli broj (integer: 3) i s njim raditi normalno, a rezultat će naravno biti pogrešan!

Tijek rada većine programa je (redom):

**UNOS (podataka, input) ⇒ OBRADA (računanje) ⇒ IZLAZ (ispis rezultata, output)**

## UNOS PODATAKA

Dva su osnovna načina unosa:

- 1.) upisivanje podataka u ćeliju worksheeta
- 2.) traženje od korisnika da unese podatke u formular za unos podataka (*inputbox*)

### 1. DOHVAT PODATAKA IZ ĆELIJE WORKSHEETA

```
k = Range("A1").Value
```

prijevod: "Neka varijabla **k** bude ono što je upisano u ćeliju **A1**".

Pošto je prethodno aktiviran "Sheet1", podrazumijeva se da se ćelija "A1" nalazi u tom worksheetu. Ukoliko se želi dohvatiti vrijednost ćelije iz nekog drugog sheeta, piše se:

```
k = Worksheets("Sheet2").Range("A1").Value
```

Drugi način definiranja položaja ćelije (ili skupa ćelija koje tvore pravokutnik):

```
k = Range(Cells(1, 1), Cells(1, 1)).Value (ćelija "A1")
```

Prva označena ćelija je gornja lijeva, a druga je donja desna. Oznaka položaja svake ćelije sastoji se od dva parametra, prvi je redni broj RETKA, a drugi je redni broj STUPCA (stupac "A" je 1, "B" je 2 ...).

Čak i kada se radi samo o jednoj ćeliji dobro je definirati obje ćelije, koje su u tom slučaju jednake.

Ovim načinom zapisivanja omogućeno je povezivanje adrese ćelije sa varijablom u programu jer umjesto brojčanih oznaka stupca i retka možemo koristiti varijable, pa čak i matematičke izraze.

### 2. FORM(ULAR) ZA UNOS PODATAKA – INPUTBOX

```
k = Inputbox("Molim unesite jedan broj")
```

prijevod: "Neka varijabla **k** bude ono što korisnik upiše u formular".

Tekst upita je proizvoljan i ovisi o onome što tražite od korisnika; bitno je da bude precizan.

# OBRADA PODATAKA

Obuhvaća sve matematičke i logičke operacije koje možemo izvršiti na i sa unesenim podacima kako bi smo dobili željeni rezultat. Osim osnovnih matematičkih operacija, često su nam potrebne konstrukcije – petlje FOR-NEXT , IF-THEN , WHILE itd.

## 1. ARITMETIČKE OPERACIJE

```
Dim n As Long (deklariranje nove varijable)
Dim k As Long (deklariranje nove varijable)
k = Worksheets("Sheet2").Range("A1").Value
n = (k + 1) ^ 2
```

U ovom slučaju program će pročitati "k" iz ćelije "A1", dodati mu "1" i rezultat kvadrirati. Mogu se koristiti operatori standardnih matematičkih operacija (+ - / \*), te funkcije potenciranja (*broj^potencija*) i vađenja kvadratnog korijena *SQR(broj)*.

Vađenje svih drugih korijena radi se upotrebom funkcije potenciranja na razlomak ( $8^{1/3} = 2$ ).

$a = 19 \setminus 6$  Ova operacija daje cjelobrojni kvocijent:  $19 \setminus 6 = 3$

$a = 19 \text{ Mod } 6$  Ova operacija daje ostatak od cjelobrojnog dijeljenja:  $19 \text{ Mod } 6 = 1$

Dakle  $19 / 6 = 3$  i ostatak 1

Treba razlikovati znak dijeljenja "/" od znaka cjelobrojnog dijeljenja "\" (Alt Gr + Q)

## 2. FOR-NEXT PETLJA (brojač)

```
Dim a As Integer (deklariranje nove varijable)
For a = 1 To 10 u prijevodu znaci: "Neka (varijabla) a ima vrijednost redom od 1 do 10"
```

Dodatak naredbi FOR koji omogućava određivanje redoslijeda brojanja je STEP:

```
For a = 1 To 10 Step 2 (samo neparne brojeve [1,3,5,7,9])
```

```
For a = 2 To 10 Step 2 (samo parni brojevi [2, 4, 6, 8 ,10])
```

```
For a = 10 To 1 Step -1 (brojanje unatrag [10, 9, 8,...])
```

```
Next a
```

Završetak svake FOR-NEXT petlje. U prijevodu znači "Vrati se na FOR i uzmi sljedeću po redu vrijednost a, zatim s njom ponovi sve operacije između FOR i NEXT".

Ukoliko programiranje zahtijeva petlju u petlji, valja paziti na redoslijed naredbi NEXT, koji je obrnut od redoslijeda slaganja naredbi FOR:

```
For a = 1 To 10
    For b = 1 To 10
        (ovdje se nalaze naredbe koje u sebi sadrže varijable a i b)
    Next b
Next a
```

### 3. IF-THEN NAREDBA GRANANJA

If a < 10 Then

U prijevodu znaci: "Ako je varijabla a manja od 10, neka se izvršavaju naredbe koje slijede"

ElseIf a = 10 Then

"Ako je varijabla a jednaka 10, neka se izvršavaju naredbe koje slijede"

Ovo je drugi uvjet, a na isti način može se nanizati jedan za drugim više uvjeta Elseif.

Else

"U svim preostalim slučajevima neka se izvršavaju naredbe koje slijede"

Ovo je posljednji uvjet, a obuhvaća sve preostale mogućnosti.

End If (Završetak IF-THEN naredbe.)

**OPERATORI USPOREĐIVANJA** su: =, >, <, >=, <=, <>, a više uvjeta moguće je i povezivati **LOGIČKIM OPERATORIMA**. Uz ostale, najčešće su u upotrebi *AND* i *OR*.

If a < 10 AND a > 5 Then (...kada je a manji od 10 **I** veći od 5 onda...)

If a < 10 OR a = 15 Then (...kada je a manji od 10 **ILI** je a jednak 15 onda...)

## IZLAZ REZULTATA (ispis)

Dva su osnovna načina ispisa rezultata rada programa:

- 1.) ispisivanje rezultata u ćeliju worksheeta
- 2.) ispisivanje rezultata izravno u prozorčiću na ekranu (*msgbox*)

### 1. ISPISIVANJE REZULTATA U ĆELIJE WORKSHEETA

```
Worksheets("Sheet1").Range("A1").Value = k
```

prijevod: "Neka se u ćeliju **A1** upiše vrijednost koju ima varijabla **k**".

Kao što postoji i drugi način definiranja položaja ćelije (ili skupa ćelija koje tvore pravokutnik) pri unosu podataka, tako isti način možemo koristiti i za ispis:

```
Worksheets("Sheet1").Activate (dio reference koji definira radni worksheet, v. napomenu)
```

```
Range(Cells(1, i), Cells(2, i)).Value = k
```

Ovim načinom zapisivanja omogućeno je povezivanje adrese ćelije sa varijablom u programu.

Primjerice, izraz koji obuhvaća prvih *i* ćelija u stupcu **A**:

```
Range(Cells(1, 1), Cells(i, 1)).Value = k
```

Adresiranje više ćelija:

```
Range("A1:A7").Value = k
```

NE može se pisati:

```
Range("A1:Ai").Value = k    ❌
```

#### VAŽNA NAPOMENA:

Tijekom vježbi primjećeno je da programi ponekad neće raditi ako se referenca unutar FOR-NEXT ili IF-THEN petlji napiše u jednom retku kao npr.:

```
Worksheets("Sheet1").Range(Cells(i,1),Cells(i,1)).Value = "patak"
```

U tom slučaju referencu treba napisati pomoću naredbe *Activate*:

```
Worksheets("Sheet1").Activate
```

```
Range(Cells(i,1),Cells(i,1)).Value = "patak"
```

### 2. ISPIS REZULTATA U PROZORČIĆU NA EKRANU – MSGBOX (message box)

```
Msgbox "Rezultat je: " & k
```

Na ekranu se pojavljuje prozorčić sa porukom "Rezultat je: (vrijednost varijable k)"

Znak "&" (Shift + 6) služi za povezivanje (nizanje) teksta i varijable u jednu rečenicu. Varijabla se ne smije stavljati pod "" jer Excel sve što je pod navodnim znacima ispisuje doslovno onako kako je upisano.

## POLJA

Polje se može zamisliti kao varijabla koja u sebi sadrži više različitih vrijednosti. Polje može biti **jednodimenzionalno** (niz varijabli) i **dvodimenzionalno** (nalik matrici).

Obična varijabla može se zamisliti kao **nul-dimenzionalno** polje.

Prilikom definiranja polja mora mu se odrediti najveća moguća veličina, a unos vrijednosti najjednostavnije je napraviti korištenjem FOR-NEXT petlje. Unesene vrijednosti su memorirane, ali ukoliko to izričito nije naređeno, nigdje se ne ispisuju.

### 1D-POLJE

```
Dim niz(1 To 50) As Single
```

Ovime je definirano 1D polje *niz* sa max 50 vrijednosti (ne moraju sve vrijednosti biti popunjene).

Unos *m* vrijednosti pomoću For-Next petlje (uz ispis u worksheet):

```
For i = 1 To m
    niz(i) = InputBox(" ")
    Worksheets("Sheet1").Activate
    Range(Cells(i,1),Cells(i,1)).Value = niz(i)
Next i
```

Pojedinoj vrijednosti u polju pristupa se pomoću indeksa u zagradi, npr.:

```
Range("A1").Value = niz(5)
```

ispisuje vrijednost 5. člana polja u ćeliju A1

### 2D-POLJE

```
Dim matrica(1 To 5, 1 To 5) As Single
```

Ovime je definirano 2D polje pod nazivom *matr* max veličine 5 x 5.

Unos  $m*n$  vrijednosti pomoću dvostruke For-Next petlje (uz ispis u worksheet):

```
For i = 1 To m
    For j = 1 To n
        matrica(i, j) = InputBox(" ")
        Worksheets("Sheet1").Activate
        Range(Cells(i,j),Cells(i,j)).Value = matrica(i,j)
    Next j
Next i
```

Element polja u ovom slučaju označava se sa dva indeksa:

```
matrica(i, j)
```

---

'Komentar

Opaske koje programer radi lakšeg kasnijeg snalaženja piše u sam program počinju znakom '. Sve što se nalazi iza tog znaka Excel ne čita.

```
End Sub
```

Kraj programa. Excel automatski dodaje ovu liniju.