

Σημειώσεις Εισαγωγής στον Προγραμματισμό σε Pascal

05 – Σταθερές και χρήση τους.

- Σταθερές, Ορισμός και Ιδιότητες:
 - Θέσεις στην μνήμη του Η/Υ στις οποίες δίνουμε όνομα και μία αρχική τιμή.
 - Σε αντίθεση με τις μεταβλητές, η τιμή μιας σταθεράς δεν μπορεί να μεταβληθεί κατά την εκτέλεση ενός προγράμματος.
- Ονομασία σταθερών.
 - Τα ονόματα των σταθερών ακολουθούν τους ίδιους κανόνες με αυτά των μεταβλητών:
 - Πρώτος χαρακτήρας ονόματος : υποχρεωτικά γράμμα.
 - Επόμενοι χαρακτήρες : γράμμα, αριθμός ή `_` (underscore).
 - Ένας ακόμα ανεπίσημος κανόνας είναι η χρήση κεφαλαίων στα ονόματα σταθερών. Αυτό κάνει τις σταθερές να ξεχωρίζουν από τις μεταβλητές μέσα στο πρόγραμμα.
 - *Παράδειγμα*: `PI` αντί για `pi`, `G` αντί για `g`, `VAT` αντί για `vat`.
- Τμήμα δήλωσης σταθερών.

Όπως και με τις μεταβλητές, πριν χρησιμοποιήσουμε μια σταθερά πρέπει να την δηλώσουμε στο τμήμα δήλωσης σταθερών. Η διαφορά είναι ότι για τις σταθερές δεν ορίζουμε όνομα και τύπο, αλλά όνομα και τιμή.

 - Όπως αναφέρθηκε παραπάνω, για να δηλώσουμε μία σταθερά πρέπει να ορίσουμε:
 - Όνομα Σύμφωνα με τους κανόνες ονομασίας που αναφέρθηκαν.
 - Τιμή Η τιμή καθορίζει και σε τι τύπου δεδομένων ανήκει η σταθερά.
 - Σύνταξη:

```
const
    ΟΝΟΜΑ = Τιμή;
```
 - *Παράδειγμα δήλωσης σταθερών :*

```
const
    PI    = 3.14159;
    VAT   = 0.24;
    COMPANY = 'IBM';
```

Η δήλωση μιας σταθεράς δεσμεύει χώρο στη μνήμη του Η/Υ. Η διαφορά με τις μεταβλητές είναι ότι ο χώρος αυτός 'κλειδώνει' μετά την δήλωση και τα περιεχόμενά του δεν μπορούν να μεταβληθούν. Αυτό σημαίνει ότι:

 - Δεν μπορούμε να χρησιμοποιήσουμε την εντολή εισόδου `read` για να εισάγουμε την τιμή μιας σταθεράς.
 - Δεν μπορούμε να εκχωρήσουμε καινούργια τιμή σε μια σταθερά (να έχουμε μια σταθερά στο αριστερό μέρος μιας εντολής εκχώρησης τιμής).
 - *Παράδειγμα :* Οι εντολές στο κύριο μέρος του παρακάτω προγράμματος **δεν είναι αποδεκτές**:

```
program Not_Valid;
const
    VAT = 0.24;
begin
    VAT := 0.40; // Προσπάθεια αλλαγής της τιμής μιας σταθεράς.
    readln(VAT); // Προσπάθεια εισαγωγής νέας τιμής από το πληκτρολόγιο.
end.
```

- Χρήση σταθερών σε ένα πρόγραμμα.

Τις σταθερές τις χρησιμοποιούμε συνήθως μέσα σε πράξεις (στο δεξί μέρος μίας εντολής εκχώρησης τιμής). Αν και μπορούμε, σπάνια τυπώνουμε την τιμή τους ως έχει.

– *Παράδειγμα* : Παρατηρούμε ότι οι σταθερές είναι με κεφαλαία.

```
→ area := PI * r * r;           // Βλέπε και Άσκηση 10
→ tot_val := net_val * VAT;      // Όπου π.χ. VAT = 0.24
```

- Πλεονεκτήματα χρήσης σταθερών.

Αν χρησιμοποιηθούν σωστά, οι σταθερές παρέχουν σημαντικά πλεονεκτήματα σε ένα πρόγραμμα:

- Εμποδίζουν την κατά λάθος μεταβολή δεδομένων και κατά συνέπεια στον υπολογισμό λάθος αποτελεσμάτων. Στα παραπάνω παραδείγματα, αν το PI δηλωθεί ως μεταβλητή, η τιμή του μπορεί κατά λάθος να μεταβληθεί (Π.χ. η εντολή `PI:=4.14`; είναι πια δυνατή αν το PI είναι μεταβλητή). Τέτοια αλλαγή θα οδηγήσει στον λάθος υπολογισμό του εμβαδού κύκλου.
- Κάνουν το πρόγραμμα παραμετροποιήσιμο. Αν η σταθερά χρησιμοποιείται σε πολλά σημεία, μία αλλαγή στην τιμή της επιφέρει αλλαγές σε όλο το πρόγραμμα. Στο παρακάτω παράδειγμα η τιμή του VAT χρησιμοποιείται δύο φορές (υπογραμμισμένα):

```
program Tax1;
const
    VAT = 0.24;
var
    NetVal1,NetVal2:real;
    TotVal1,TotVal2:real;
begin
    write('1st order price:');
    readln(NetVal1);
    write('2nd order price:');
    readln(NetVal2);

    TotVal1 := NetVal1 + NetVal1 * VAT;
    TotVal2 := NetVal2 + NetVal2 * VAT;

    writeln('1st order total value:',TotVal1:8:2);
    writeln('2nd order total value:',TotVal2:8:2);
end.
```

Μία απλή αλλαγή της VAT στο τμήμα δήλωσης (π.χ. από 0.24 σε 0.20) αυτόματα θα αλλάξει και την τιμή της στις εντολές εκχώρησης. Χωρίς τη χρήση σταθερών θα έπρεπε να αντικαταστήσουμε τη νέα τιμή όπου υπάρχει.

Π.χ. Θα έπρεπε να αλλάξουμε και τις δύο εντολές από:

```
TotVal1 := NetVal1 + NetVal1 * 0.24;
TotVal2 := NetVal2 + NetVal2 * 0.24;
```

Σε:

```
TotVal1 := NetVal1 + NetVal1 * 0.20;
TotVal2 := NetVal2 + NetVal2 * 0.20;
```