

Αντικειμενοστρεφής Προγραμματισμός II

1^ο φυλλάδιο εργαστηρίου – 14 & 16/10/2024

Πέρασμα παραμέτρων κατά την εκτέλεση του προγράμματος

Κατά την εκτέλεση του προγράμματος μπορούμε να περάσουμε παραμέτρους στην `main()` οι οποίες αποθηκεύονται στον πίνακα `args`. Οι παράμετροι συντάσσονται, σε σειρά, μετά το όνομα του προγράμματος όταν αυτό εκτελείται στη γραμμή εντολών.

Ένα πρώτο παράδειγμα χρήσης παραμέτρων

Με τη χρήση του `netbeans` δημιουργείστε ένα αρχείο με όνομα `Identity.java` το οποίο θα περιέχει τον παρακάτω πρόγραμμα σε `java`.

```
public class Identity {  
    public static void main(String args[]) {  
        System.out.println("My name is ");  
        System.out.println(args[0] + " " + args[1]);  
        System.out.println(" and i am ");  
        System.out.println(args[2]);  
        System.out.println(" years old.");  
    }  
}
```

Εκτελέστε το παραπάνω πρόγραμμα ορίζοντας ως παραμέτρους τις τιμές «Iwannis», «Doukas» και 25 (Run -> Set Project Configuration -> Customize, Κατηγορία : Run -> Arguments). Σε ποιο συμπέρασμα καταλήγετε μετά την εκτέλεση του προγράμματος; Εκτελέστε το ίδιο πρόγραμμα χωρίς τις παραπάνω παραμέτρους.

Αλλάξτε τις εντολές εκτύπωσης του παραδείγματος, έτσι ώστε να τυπώνονται όλες οι πληροφορίες σε μία γραμμή.

Διαμορφώστε το σώμα της μεθόδου ως εξής:

```
public static void main(String args[]) {  
    if (args.length == 3) {  
        System.out.print("My name is ");  
        System.out.print(args[0] + " " + args[1]);  
        System.out.print(" and i am ");  
        System.out.print(args[2]);  
        System.out.println(" years old.");  
    }  
    else  
        System.out.println("Missing parameters");  
}
```

Μεταφράστε ξανά το πρόγραμμα και εκτελέστε το με τον ίδιο τρόπο. Είναι αναμενόμενα τα αποτελέσματα;

Στη συνέχεια κάντε τις αλλαγές που απαιτούνται ώστε να τυπώνεται πρώτα η ηλικία και στη συνέχεια το ονοματεπώνυμο ενός ανθρώπου.

Πιστεύετε ότι μπορούμε να τυπώσουμε όλα τα στοιχεία ενός ανθρώπου με τη χρήση μόνο μίας εντολής; Τι αλλαγές θα κάνατε στο πρόγραμμα σας;

Ένα δεύτερο παράδειγμα χρήσης παραμέτρων

Στο παράδειγμα που ακολουθεί παρουσιάζουμε έναν απλό τρόπο για να εμφανίσουμε όλες τις παραμέτρους που περνάμε κατά την εκτέλεση του προγράμματος:

```
public class Print_CommandLine_Args {
    public static void main(String[] args) {
        System.out.println("The arguments of the program are: ");
        for (int i=0; i<args.length; i++)
            System.out.print(args[i] + " ");
    }
}
```

Προχωρήστε στη μεταγλώττιση και εκτέλεση του παραπάνω προγράμματος ορίζοντας διαφορετικές παραμέτρους (πλήθους και τύπου) κάθε φορά.

1^η εργαστηριακή άσκηση

Συμπληρώστε την ακόλουθη main μέθοδο:

```
public class Calc {
    public static void main(String args[]) {
        int operand1, operand2;
        char operator;

        if (args.length != 3) {
            System.out.println("Missing parameters ...");
            return;
        }

        operand1 = // Μετατροπή της 1ης παραμέτρου σε ακέραιο
        operand2 = // Μετατροπή της 2ης παραμέτρου σε ακέραιο

        operator = args[2].charAt(0);

        switch(operator) {
            case ('-'):
                // Εμφάνιση αποτελέσματος εφαρμόζοντας τον τελεστή -
                break;
            case ('+'):
                // Εμφάνιση αποτελέσματος εφαρμόζοντας τον τελεστή +
                break;

            // Αντίστοιχο case για τον τελεστή x
            case ('/'):
                if (operand2==0)
                    System.out.println("Unable to execute the operation");
                else {
                    // Εμφάνιση αποτελέσματος εφαρμόζοντας τον τελεστή /
                    break;
                }
            default:
                System.out.println("Error: invalid operator!");
        }
    }
}
```

Το πρόγραμμα δέχεται κατά την εκτέλεση του, τρεις παραμέτρους: 2 ακέραιες παραμέτρους και έναν αριθμητικό τελεστή (+, -, x ή /) και υπολογίζει το αποτέλεσμα της πράξης σύμφωνα με τον τελεστή που έχει ορισθεί.

Αν για παράδειγμα εκτελέσουμε το πρόγραμμα με παραμέτρους 2 8 + θα πρέπει να εμφανίζει :

2 + 8 = 10

Εισαγωγή δεδομένων στο πρόγραμμα από το πληκτρολόγιο

Υπάρχουν δύο κλασικοί τρόποι εισαγωγής δεδομένων (input) σε ένα πρόγραμμα Java: είτε με την χρήση εντολών από το standard input, είτε με την ευκολία που προσφέρει το γραφικό περιβάλλον της Java με την βοήθεια των βιβλιοθηκών AWT, Swing και JavaFX. Στο ακόλουθο παράδειγμα θα ασχοληθούμε με την ανάλυση της κλάσης **Scanner** η οποία χρησιμοποιείται για την εισαγωγή δεδομένων από το πληκτρολόγιο χωρίς τη χρήση γραφικού περιβάλλοντος.

Από την έκδοση 5.0 της Java, έχει συμπεριληφθεί μια ειδική κλάση με την ονομασία Scanner. Χρησιμοποιώντας αυτή την κλάση, είναι πολύ εύκολο να γράψουμε ένα πρόγραμμα στο οποίο ο χρήστης θα μπορεί να εισάγει τα δικά του δεδομένα σε αυτό χρησιμοποιώντας το πληκτρολόγιο του. Η κλάση Scanner, είναι μέρος ενός μεγαλύτερου πακέτου με την ονομασία util. Για να είμαστε σε θέση να χρησιμοποιήσουμε την κλάση Scanner σε κάθε πρόγραμμα θα πρέπει στην αρχή του κώδικα μας να προσθέσουμε το εξής:

```
import java.util.*; ή εναλλακτικά
```

```
import java.util.Scanner;
```

Με τον αστερίσκο (*) λέμε ότι όλες οι κλάσεις του συγκεκριμένου πακέτου (util) μπορεί να χρησιμοποιηθούν στον κώδικα μας, και έτσι ο μεταγλωττιστής θα τις συμπεριλάβει κατά τη διαδικασία της μεταγλώττισης. Με αυτό τον τρόπο, η κλάση Scanner και όλες οι μέθοδοι της είναι διαθέσιμες προς χρήση στο πρόγραμμα μας.

Ακολουθεί χαρακτηριστικό παράδειγμα χρήσης της κλάσης Scanner:

```
import java.util.Scanner;
```

```
public class Addition {  
    public static void main(String[] args) {  
        int number1, number2;  
        int result;  
  
        Scanner input=new Scanner(System.in);  
  
        System.out.print("Enter first integer : ");  
        number1 = input.nextInt();  
  
        System.out.print("Enter second integer : ");  
        number2 = input.nextInt();  
  
        result = number1 + number2;  
        System.out.println("Sum is : " + result );  
    }  
}
```

Με αυτό τον τρόπο δημιουργείται ένα νέο αντικείμενο της κλάσης Scanner το οποίο ονομάσαμε input. Το System.in, που βρίσκεται στην παρένθεση, αντιπροσωπεύει το πληκτρολόγιο. Η κλάση Scanner, έχει αρκετές μεθόδους εισαγωγής δεδομένων, κάθε μια εκ των οποίων συνεργάζεται και με διαφορετικό τύπο δεδομένων ανάλογα με τη μεταβλητή στην οποία θα εισαχθούν τα δεδομένα (*nextFloat()*, *nextDouble()*, κλπ.)

2^η εργαστηριακή άσκηση

Να υλοποιήσετε ένα πρόγραμμα στην Java το οποίο θα διαβάζει επαναληπτικά ακέραιους αριθμούς από το πληκτρολόγιο μέχρι να δοθεί η τιμή 0. Στο τέλος θα εμφανίζεται το πλήθος των αριθμών που διαβάστηκαν συνολικά καθώς και πόσοι από αυτούς είναι περιττοί και πόσοι άρτιοι.

Πλαίσια Διαλόγου για εισαγωγή δεδομένων

Η Java όπως και άλλες γλώσσες προγραμματισμού παρέχει τη δυνατότητα δημιουργίας μιας σειράς διαλόγων που ονομάζονται **κοινοί διάλογοι** (common dialogs) και επιτρέπουν την εισαγωγή δεδομένων και την προβολή μηνυμάτων σε γραφικό περιβάλλον. Η κλάση **JOptionPane** που ανήκει στο πακέτο javax.swing περιέχει μια σειρά από static μεθόδους για τη δημιουργία πλαισίων διαλόγου. Το πακέτο **javax.swing** περιέχει κλάσεις για τη δημιουργία παραθυρικών εφαρμογών και γραφικών. Αναλυτικά η δημιουργία εφαρμογών με γραφικό περιβάλλον θα μας απασχολήσει σε επόμενα φυλλάδια.

Η JOptionPane περιέχει τη μέθοδο **showInputDialog** που στην απλή της μορφή δέχεται ως όρισμα μια συμβολοσειρά/μήνυμα που εμφανίζει σε πλαίσιο διαλόγου και επιστρέφει μια συμβολοσειρά που είναι η απάντηση του χρήστη:

```
import javax.swing.JOptionPane;

public class Dialog1 {
    public static void main(String[] args) {
        String courseGrade;

        while (true){
            courseGrade = JOptionPane.showInputDialog("Ποιος είναι ο βαθμός σου στην Java;");
            if (Integer.parseInt(courseGrade) >=0 &&
                Integer.parseInt(courseGrade) <=10)
                break;
        }
        System.out.println("Ο βαθμός " + courseGrade + " είναι έγκυρος!");
    }
}
```

Το παραπάνω πρόγραμμα διαβάζει επαναληπτικά από τον χρήστη την βαθμολογία σε ένα μάθημα και η διαδικασία ολοκληρώνεται όταν η καταχώριση του χρήστη είναι έγκυρη.

Η στατική μέθοδος **showMessageDialog** που ανήκει και αυτή στην JOptionPane χρησιμοποιείται για την εμφάνιση του μηνύματος που δέχεται ως παράμετρο, σε κατάλληλο πλαίσιο διαλόγου:

```
import javax.swing.JOptionPane;

public class Dialog2 {
    public static void main(String args[]){
        JOptionPane.showMessageDialog(null, "Fun ways to learn JAVA!");
    }
}
```

```
}  
}
```

Στην πρώτη παράμετρο της μεθόδου ορίζουμε το στοιχείο προβολής μπροστά από το οποίο θα εμφανιστεί το πλαίσιο διαλόγου. Αν αυτή η παράμετρος είναι null το πλαίσιο διαλόγου στοιχίζεται στο κέντρο της οθόνης.

3^η εργαστηριακή άσκηση

Να υλοποιήσετε ένα πρόγραμμα στην Java το οποίο θα διαβάζει τρεις ακέραιους αριθμούς και θα υπολογίζει τον μέγιστο. Η είσοδος των δεδομένων αλλά και η εμφάνιση των αποτελεσμάτων θα πρέπει να γίνει με κατάλληλα πλαίσια διαλόγου.

Τύποι δεδομένων και μετατροπές στην Java

Όταν εκτελούμε πράξεις στην Java μεταξύ δύο ανόμοιων τύπων (π.χ. ενός ακέραιου και double) ο κανόνας που ισχύει είναι ότι ο μικρότερος από τους δυο τύπους μετατρέπεται αυτόματα στον μεγαλύτερο, οπότε η πράξη εκτελείται τελικά μεταξύ όμοιων τύπων:

```
long l = 10L; // Οι ακέραιες κυριολεκτικές τιμές είναι εξ ορισμού int  
// ενώ οι δεκαδικές κυριολεκτικές τιμές είναι εξ ορισμού double  
double d = 12.1;  
System.out.println(l + d); // Το αποτέλεσμα είναι double: 22.1
```

4^η εργαστηριακή άσκηση

Εντοπίστε ποιες πράξεις από τις ακόλουθες θα εκτελεστούν με επιτυχία και σε ποιες θα εμφανισθεί λάθος κατά την μεταγλώττιση.

```
byte b1 = 127;  
byte b2 = 128;  
char c = b1 + b2;  
double d = b2 + c;  
int i = d;  
float f = 3.5;
```