



Ελληνική Δημοκρατία
Τεχνολογικό Εκπαιδευτικό
Ίδρυμα Ηπείρου

Προγραμματισμός Διαδικτύου

Ενότητα 7 : Αρχεία στην Java

Ιωάννης Τσούλος



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Τμήμα Μηχανικών Πληροφορικής Τ.Ε

Προγραμματισμός Διαδικτύου

Ενότητα 7 : Αρχεία στην Java

Ιωάννης Τσούλος
Επίκουρος Καθηγητής
Άρτα, 2015



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.





Χρηματοδότηση

- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «**Εκπαίδευση και Δια Βίου Μάθηση**» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Η κατηγορία `InputStream`

- Αφηρημένη κατηγορία και χρησιμοποιείται για την ανάγνωση δεδομένων στην Java από αρχεία εισόδου.
 - `int read()`: Επιστρέφει με την μορφή ακεραίου τον επόμενο χαρακτήρα στην είσοδο. Αν τα δεδομένα έχουν τελειώσει τότε επιστρέφει -1.
 - `int read(byte[] b)`: Διαβάζουμε από την είσοδο έναν πίνακα δεδομένων που έχει μέγιστο μέγεθος `b.length`. Αν τα δεδομένα έχουν τελειώσει επιστρέφει -1.
 - `int available()`: Επιστρέφει το πλήθος των συμβόλων που απομένουν για ανάγνωση στο αρχείο.
 - `long skip(long count)`: Παρακάμπτει `count` στοιχεία από την είσοδο.
 - `close()`: Κλείνει το αρχείο.



Η κατηγορία InputStream

Ανάγνωση αλφαριθμητικών απο το πληκτρολόγιο

```
import java.io.*;
public class File1
{
    public static void main(String arg[])
    {
        byte line[]=new byte[160];
        try
        {
            System.in.read(line);
            String s=new String(line);
            System.out.println("Data_=_"+s);
        }catch(IOException e)
        {
            System.out.println (
                "Something_is_wrong_with_the_input");
        }
    }
}
```



Η κατηγορία `FileInputStream`

- Απογονική κατηγορία της `InputStream` και χρησιμοποιείται για την ανάγνωση δεδομένων από φυσικά αρχεία στον δίσκο.

```
import java.io.*;
public class File2
{
    public static void main(String arg[])
    {
        try
        {
            FileInputStream file=new
                FileInputStream("test.dat");
            boolean flag=false;
            while(!flag)
            {
                int input=file.read();
                if(input!=-1)
                {
                    flag=true;
                }
                else
                {
                    char ch=(char)input;
                    System.out.print(ch);
                }
            }
            file.close();
        }
        catch(IOException e)
        {
            System.out.println("Problem_with_input");
        }
    }
}
```



Η κατηγορία `OutputStream`

- Αφηρημένη κατηγορία που χρησιμοποιείται για την εγγραφή δεδομένων σε αρχεία εξόδου ή ακόμα και σε συσκευές όπως η οθόνη.
 - `void write(int b)`: Γράφει τον χαρακτήρα `b` στην έξοδο.
 - `void write(byte[] b)`: Γράφει τους χαρακτήρες που είναι αποθηκευμένοι στον πίνακα `b`
 - `void flush()`: Αδειάζει την έξοδο.
 - `void close()`: Κλείνει το αρχείο



Η κατηγορία FileOutputStream

- Η κατηγορία FileOutputStream χρησιμοποιείται για την εγγραφή δεδομένων σε φυσικά αρχεία που βρίσκονται στον δίσκο.
- Το πρόγραμμα του αλγορίθμου ανοίγει πάλι το αρχείο test.dat και αντιγράφει τους χαρακτήρες που βρίσκονται σε αυτό στο αρχείο newtest.dat

```
import java.io.*;
public class File3
{
    public static void main(String arg[])
    {
        try
        {
            FileInputStream in=new FileInputStream("test.dat");
            FileOutputStream out=new FileOutputStream("newtest.dat");
            while(true)
            {
                int input=in.read();
                if(input==-1) break;
                out.write(input);
            }
            in.close();
            out.close();
        }
        catch(IOException e)
        {
            System.out.println("Problem_with_the_files");
        }
    }
}
```



Αρχεία δεδομένων

■ Οι κατηγορίες `DataInputStream` και `DataOutputStream`

- Η κατηγορία `DataInputStream` χρησιμοποιείται για την ανάγνωση απλών δεδομένων από αρχεία εισόδου.
- Στην συνάρτηση δημιουργίας της παίρνει σαν όρισμα ένα αντικείμενο της κατηγορίας `FileInputStream`.
 - `char readChar()`: Διαβάζει και επιστρέφει έναν χαρακτήρα.
 - `int readInt()`: Διαβάζει και επιστρέφει έναν ακέραιο.
 - `float readFloat()`: Διαβάζει και επιστρέφει έναν απλό δεκαδικό αριθμό.
 - `double readDouble()`: Διαβάζει και επιστρέφει έναν `double` δεκαδικό αριθμό.
 - `String readUTF()`: Διαβάζει και επιστρέφει ένα αλφαριθμητικό σε ₁₀ πρότυπο UTF



Αρχεία δεδομένων

- Παρόμοια με την κατηγορία `DataInputStream` υπάρχει και η κατηγορία `DataOutputStream` που χρησιμοποιείται για την αποθήκευση δεδομένων σε κάποιο αρχείο.
 - `void writeChar(char)`: Αποθηκεύει έναν χαρακτήρα στο αρχείο.
 - `void writeInt(int)`: Αποθηκεύει έναν ακέραιο στο αρχείο.
 - `void writeFloat(float)`: Αποθηκεύει έναν απλό δεκαδικό στο αρχείο.
 - `void writeDouble(double)`: Αποθηκεύει έναν `double` αριθμό στο αρχείο.
 - `void writeUTF(String)`: Αποθηκεύει ένα αλφαριθμητικό σε πρότυπο UTF στο αρχείο.



Αρχεία δεδομένων

```
import javax.swing.*;
import java.io.*;
public class File4
{
    public static int readInt()
    {
        String s=JOptionPane.showInputDialog("Enter an integer_");
        int x=Integer.parseInt(s);
        return x;
    }
    public static void main(String arg[])
    {
        try
        {
            int n,x,i;
            FileOutputStream out=new
                FileOutputStream("numbers.dat");
            DataOutputStream st=new DataOutputStream(out);
            n=readInt();
            st.writeInt(n);
            for (i=1;i<=n;i++)
            {
                x=readInt();
                st.writeInt(x);
            }
            st.close();
        }
        catch(IOException e)
        {
            System.out.println("Problem_with_the_output_file");
        }
    }
}
```



Αρχεία δεδομένων

■ Ανάγνωση κειμένου

- Τα προηγούμενα παραδείγματα μπορούν να χρησιμοποιηθούν για να διαβάσουμε δεδομένα σε δυαδική μορφή αλλά δεν μπορούν να χρησιμοποιηθούν για να διαβάσουμε δεδομένα σε απλή μορφή.

```
import java.io.*;
public class File6
{
    public static void main(String arg [])
    {
        try
        {
            FileInputStream out=new
                FileInputStream ("numbers.txt");
            DataInputStream st=new DataInputStream(out)
            int n,x,i ,sum=0;
            n=Integer.parseInt (st.readLine ());
            for (i=1;i<=n;i++)
            {
                x=Integer.parseInt (st.readLine ());
                sum=sum+x;
            }
            System.out.println ("Average_=_"+(sum/n));
            st.close ();
        }
        catch (IOException e)
        {
            System.out.println ("Error_in_input_");
        }
    }
}
```



Η κατηγορία File

- Χρησιμοποιείται όχι για να διαβάσουμε ή να γράψουμε δεδομένα σε ένα αρχείο αλλά για να λάβουμε πληροφορίες για αυτό ή για πιο σύνθετες εργασίες όπως διαγραφή του αρχείου.
 - `File(String name)`: Χρησιμοποιεί το `name` σαν όνομα του αρχείου
 - `File(String directory,String name)`: Χρησιμοποιεί το `directory` σαν φάκελο μέσα στο οποίο θα υπάρχει το αρχείο με όνομα `name`.
 - `boolean exists()`: Επιστρέφει αληθές αν το αρχείο υπάρχει
 - `boolean isFile()`: Επιστρέφει αληθές αν το αρχείο είναι απλό αρχείο
 - `boolean isDirectory ()`: Επιστρέφει αληθές αν το αρχείο είναι κατάλογος
 - `boolean canRead()`: Επιστρέφει αληθές αν υπάρχουν δικαιώματα ανάγνωσης στο αρχείο
 - `boolean canWrite()`: Επιστρέφει αληθές αν υπάρχουν δικαιώματα εγγραφής στο αρχείο
 - `boolean delete()`: Προσπαθεί να διαγράψει το αρχείο Αν το επιτύχει επιστρέφει αληθές.
 - `String getName()`: Επιστρέφει το όνομα του αρχείου
 - `String getPath()`: Επιστρέφει το πλήρες μονοπάτι προς το αρχείο
 - `String getParent()`: Επιστρέφει το όνομα του καταλόγου στο οποίο βρίσκεται το αρχείο
 - `String[] list`: Επιστρέφει τα ονόματα των αρχείων του καταλόγου (αν φυσικά το αρχείο είναι κατάλογος)
 - `long length()`: Επιστρέφει το πλήθος των bytes στο αρχείο και 0 αν είναι κατάλογος
 - `long lastModified ()`: Επιστρέφει την τελευταία ημερομηνία αλλαγής του αρχείου



Η κατηγορία File

Περιορισμένη υλοποίηση του προγράμματος ls του λειτουργικού συστήματος Linux

```
import javax.swing.*;
import java.io.*;
public class File7
{
    public static void showDetails(String name)
    {
        File myfile=new File(name);
        System.out.print(name+"\t");
        long size=myfile.length();
        System.out.print(size+"\t");
        if(myfile.isDirectory()) System.out.print("d");
        else System.out.print("-");
        if(myfile.canRead()) System.out.print("r");
        else System.out.print("-");
        if(myfile.canWrite()) System.out.print("w");
        else System.out.print("-");
        System.out.println("");
    }
    public static void main(String arg[])
    {
        String name=JOptionPane.showInputDialog("Enter_directory")
        File myfile=new File(name);
        if(!myfile.exists())
            System.out.println("File_not_readable");
        else
        {
            if(myfile.isFile()) showDetails(name);
            else
            {
                int i;
                String [] list=myfile.list();
                for(i=0;i<list.length;i++)
            }
        }
    }
}
```



Βιβλιογραφία

- Πλήρες Εγχειρίδιο της Java 2 Platform, Laura Lemay & Rogers Cadenhead, Ελληνική έκδοση Γκιούρδα
- Java Προγραμματισμός, 6η έκδοση, Harvey M. Deitel, Paul J. Deitel . Εκδόσεις Μ. Γκιούρδας, Αθήνα 2005.
- Εισαγωγή στη γλώσσα προγραμματισμού Java, Εμμ. Σκορδαλάκης, Εργαστήριο Λογισμικού Ομάδα Τεχνολογίας Λογισμικού ΕΜΠ
- Java με UML, Else Lervik και Vegard B. Havdal, Εκδόσεις Κλειδάριθμος, 2005.
- Προγραμματισμός με Java, Γιάννη Κάβουρα, Εκδόσεις Κλειθάρθμος, Αθήνα 2003.
- Java in a nutshell, Deluxe Edition και Java Examples in a Nutshell, David Flanagan, Εκδόσεις OReilly
- Java Collections An Introduction to ADTs, Data Structures and Algorithms, David Watt, Deryck Brown, John Wiley and Sons
- Thinking in Java (3rd edition), B. Eckel, Pearson Prentice Hall, 2003.
- Introduction to Java Programming (5th edition), Y. Daniel Liang, Pearson Prentice Hall, 2005.



Σημείωμα Αναφοράς

Copyright Τεχνολογικό Ίδρυμα Ηπείρου. Ιωάννης Τσούλος.
Προγραμματισμός Διαδικτύου.

Έκδοση: 1.0 Άρτα, 2015. Διαθέσιμο από τη δικτυακή
διεύθυνση:

<http://eclass.teiep.gr/OpenClass/courses/COMP114/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού-Μη Εμπορική Χρήση-Όχι Παράγωγα Έργα 4.0 Διεθνές [1] ή μεταγενέστερη. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, Διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/deed.el>



Τέλος Ενότητας

Επεξεργασία: Ευάγγελος Καρβούνης
Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Τέλος Ενότητας

Αρχεία στην Java



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

