

Δομές Δεδομένων & Ανάλυση Αλγορίθμων 3ο Εξάμηνο

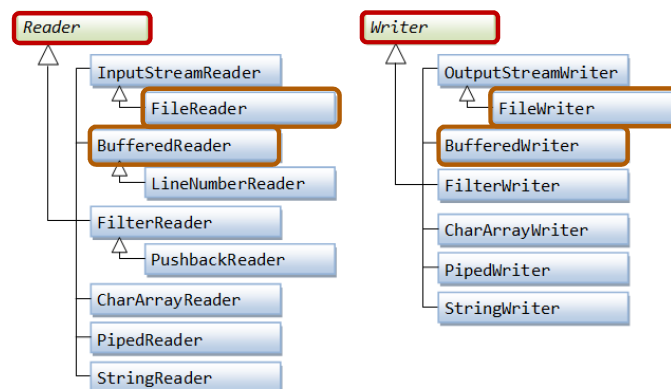
• Αρχεία Κειμένου (Text Files)

Δημοσθένης Σταμάτης

<http://www.iee.ihu.gr/~demos>

Τμήμα Μηχανικών Πληροφορικής & Ηλεκτρονικών Συστημάτων

Βασικές κλάσεις για την επεξεργασία αρχείων κειμένου (package java.io)



Βασικές κλάσεις για την επεξεργασία αρχείων κειμένου (package java.io)

```
java.lang.Object
|
+--java.io.Reader
|
+--java.io.BufferedReader
|
+--java.io.InputStreamReader
|
+--FileReader
```

```
InputStreamReader ISR = new InputStreamReader (System.in);
BufferedReader Keyboard = new BufferedReader (ISR);
```

Βασικές κλάσεις για την επεξεργασία αρχείων κειμένου (package java.io)

```
java.lang.Object
|
+--java.io.Writer
|
+--java.io.BufferedWriter
|
+--java.io.OutputStreamWriter
|
+--FileWriter
```

Βασικές Μέθοδοι της κλάσης Reader

Μέθοδος	Περιγραφή
<code>int read()</code>	Διαβάζει έναν χαρακτήρα και τον επιστρέφει σαν έναν ακέραιο από 0 ως 16383. Επιστρέφει -1 εάν συναντήσει το τέλος του ρεύματος.
<code>int read(char[] cbuf)</code>	Διαβάζει χαρακτήρες και τους τοποθετεί στον πίνακα <code>cbuf</code> . Επιστρέφει τον αριθμό των bytes που διάβασε ή -1 εάν συναντήσει το τέλος του ρεύματος.
<code>long skip(long n)</code>	Προσπαθεί να παρακάμψει τους επόμενους <code>n</code> χαρακτήρες του <code>stream</code> εισόδου. Επιστρέφει τον αριθμό των χαρακτήρων που κατάφερε να παρακάμψει.
<code>abstract void close()</code>	Κλείνει το <code>stream</code> εισόδου.

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

Βασικές Μέθοδοι της κλάσης BufferedReader

Μέθοδος	Περιγραφή
<code>public int read()</code>	Διαβάζει έναν χαρακτήρα Επαναορίζει την <code>read</code> της κλάσης <code>Reader</code>
<code>public String readLine()</code>	Διαβάζει μία γραμμή κειμένου. Μία γραμμή θεωρείται ότι τελειώνει είτε από τον χαρακτήρα <code>line feed</code> (' <code>\n</code> ') είτε από τον χαρακτήρα <code>carriage return</code> (' <code>\r</code> '), ή από τους <code>carriage return</code> και <code>linefeed</code> . Επιστρέφει ένα <code>String</code> με τους χαρακτήρες της γραμμής στο οποίο δεν συμπεριλαμβάνονται οι ειδικοί χαρακτήρες αλλαγής γραμμής. Επιστρέφει <code>null</code> αν συναντήσει
<code>long skip(long numChars)</code>	Προσπαθεί να παρακάμψει τους επόμενους <code>numChars</code> χαρακτήρες του <code>stream</code> εισόδου επιστρέφοντας τον αριθμό όσων αγνόησε.
<code>void close()</code>	Κλείνει το <code>stream</code> εισόδου. Μετά το κλείσιμο προσπαθείς ανάγνωσης δεδομένων οδηγούν σε εξαίρεση IOException

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

Βασικές Μέθοδοι της κλάσης `Writer`

Μέθοδος	Περιγραφή
<code>void write(int c)</code>	Γράφει έναν χαρακτήρα στο ρεύμα εξόδου. Ο χαρακτήρας που γράφεται περιέχεται στα 16 λιγότερο σημαντικά bits της ακεραίας τιμής <code>c</code> , ενώ τα 16 περισσότερο σημαντικά αγνοούνται.
<code>void write(String str)</code>	Γράφει το <code>String str</code> στο ρεύμα εξόδου.
<code>abstract void flush()</code>	Εκκενώνει το ρεύμα εξόδου από τους χαρακτήρες που περιέχει. Εάν το ρεύμα έχει αποθηκεύσει κάποιους χαρακτήρες σε έναν <code>buffer</code> με τη βοήθεια μεθόδων <code>write()</code> , οι χαρακτήρες αυτοί γράφονται κατ' ευθείαν στον προορισμό τους. Στη συνέχεια εάν η έξοδος του ρεύματος κατευθύνεται σε ένα άλλο ρεύμα και αυτό με τη σειρά του εκκενώνεται.
<code>abstract void close()</code>	Κλείνει το <code>stream</code> εξόδου αφού πρώτα το εκκενώσει.

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

Βασικές Μέθοδοι της κλάσης `BufferedWriter`

Μέθοδος	Περιγραφή
<code>void write(int c)</code>	Γράφει έναν χαρακτήρα στο ρεύμα εξόδου.
<code>void write(String s, int i, int n)</code>	Γράφει ένα τμήμα του <code>String s</code> στο ρεύμα εξόδου. Το τμήμα αυτό του <code>String</code> αρχίζει από τη θέση <code>i</code> και αποτελείται από τους <code>n</code> επόμενους χαρακτήρες.
<code>void.newLine()</code>	Γράφει στο ρεύμα εξόδου το ειδικό <code>string</code> που αντιστοιχεί σε «αλλαγή γραμμής». Το <code>String</code> αυτό εξαρτάται από τη μηχανή/λειτουργικό σύστημα και ορίζεται στα πλαίσια του <code>system property</code> σαν <code>line.separator</code> .
<code>abstract void flush()</code>	Εκκενώνει το ρεύμα εξόδου από τους χαρακτήρες που περιέχει. Εάν το ρεύμα έχει αποθηκεύσει κάποιους χαρακτήρες σε έναν <code>buffer</code> με τη βοήθεια μεθόδων <code>write()</code> , οι χαρακτήρες αυτοί γράφονται κατ' ευθείαν στον προορισμό τους. Στη συνέχεια εάν η έξοδος του ρεύματος κατευθύνεται σε ένα άλλο ρεύμα και αυτό με τη σειρά του εκκενώνεται.
<code>abstract void close()</code>	Κλείνει το <code>stream</code> εξόδου αφού πρώτα το εκκενώσει.

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

```
public class CopyFile {
    public static void main(String[] args) throws IOException {

        File inputFile = new File("filename1.txt");
        File outputFile = new File("filename2.txt");

        FileReader in = new FileReader(inputFile);
        FileWriter out = new FileWriter(outputFile);

        int charValue;
        while ((charValue = in.read()) != -1)
            out.write(charValue);
        in.close();
        out.close();
    }
}
```

```
public class ReadSource {
    public static void main(String[] arguments) {
        try {
            FileReader file = new FileReader("ReadSource.java");
            BufferedReader buff = new BufferedReader(file);
            boolean eof = false;
            while (!eof) {
                String line = buff.readLine();
                if (line == null)
                    eof = true;
                else
                    System.out.println(line);
            }
            buff.close();
        } catch (IOException e) {
            System.out.println("Error -- " + e.toString());
        }
    }
}
```

Convert text to capital letters

```

public static void main(String[] arguments) {

    try {
        Scanner KB = new Scanner(System.in);
        System.out.println("Give me the name of the file: ");
        String sourceName= KB.next();

        // Create file objects
        File source = new File(sourceName);
        File temp = new File("cap" + sourceName + ".tmp");

        // Create input stream
        FileReader fr = new FileReader(source);
        BufferedReader in = new BufferedReader(fr);

        // Create output stream
        FileWriter fw = new FileWriter(temp);
        BufferedWriter out = new BufferedWriter(fw);
    }
}

```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ

```

        boolean eof = false;
        int inChar = 0;
        do {
            inChar = in.read();
            if (inChar != -1) {
                char outChar = Character.toUpperCase((char)inChar);
                out.write(outChar);
            } else
                eof = true;
        } while (!eof);
        in.close();
        out.close();

        boolean deleted = source.delete();
        if (deleted) temp.renameTo(source);
    }
    catch (IOException e) {
        System.out.println("Error -- " + e.toString());
    }
    catch (SecurityException se) {
        System.out.println("Error -- " + se.toString());
    }
}

```

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Γ' ΕΞΑΜΗΝΟ