

```
package analyzer;

import java.io.*;

/** * Handles the file, and has a buffer to */
public class FileLoader {

    private static final int HALFSIZE = 1024;
    private static final int SIZE = 2*HALFSIZE;
    private InputStream stream;
    private byte[] buffer;
    private int[] row;
    private int[] column;
    private boolean half;
    private int index;

    public FileLoader(File file) throws IOException {
        this.stream = new FileInputStream(file);
        this.buffer = new byte[SIZE];
        this.row = new int[SIZE];
        this.column = new int[SIZE];
        this.half = true; // to start filling the lower part
        load();
        this.index = -1;
    }

    /**
     * Closes the opened file
     */
    public void close() {
        try { this.stream.close(); }
        catch(IOException ex) {}
    }

    private void load() {
        int begin = (half? 0 : HALFSIZE);
        int read = 0;
        try { read = stream.read(buffer,begin,HALFSIZE); }
        catch(IOException ex) {}
        if(read < HALFSIZE) {
```

```
        for(int i=read; i<HALFSIZE; i++) buffer[begin+i] = 0;
    }
    int prev = (half? SIZE-1 : HALFSIZE-1);
    int newrow = row[prev];
    int newcolumn = column[prev]+1;
    if(buffer[prev] == '\n') { newrow++; newcolumn = 0; }
    for(int i=begin; i<begin+HALFSIZE; i++){
        row[i] = newrow;
        column[i] = newcolumn;
        newcolumn++;
        if(buffer[i] == '\n') { newrow++; newcolumn = 0; }
    }
    half = !half;
}

/**
 * @return Next char of the stram
 */
public char getNextChar() {
    index++;
    if(index==HALFSIZE && !half) load();
    else if(index == SIZE && half) { index=0; load(); }
    else if(index == SIZE) { index = 0; }
    return (char) buffer[index];
}

/**
 * @return current row in the file
 */
public int getRow() {
    return row[index];
}

/**
 * @return current column in the file
 */
public int getColumn() {
    return column[index];
}
```

```
/**
 * Retracts the char pointer
 * @param disp amount of slots to retract
 */
public void retract(int disp) {
    index -= disp;
    if(index <0) index += SIZE;
}
}
```