

```
import java.io.*;

public class Lue {
/*****
Lukurutiinit Johdatus ohjelmointiin -kurssille syksyllä 1997   Arto Wikla
*Lisännyt Atte Tanskanen 2004

Operaatiot:

Lue.rivi()          antaa seuraavan syöttörivin          (String)
Lue.kluku()         "      "      kokonaisluvun          (int)
Lue.dluku()         "      "      desimaaliluvun          (double)
*Lue.lluku()        "      "      kokonaisluvun          (long)
Lue.merkki()        antaa seuraavan syöttörivin ensimmäisen merkin (char)
*Lue.tavutTiedostosta() antaa tiedoston sisällön          (byte[])
*Lue.riviPuskurista() antaa tiedostopuskurin seuraavan rivin (String)

*****/
    static BufferedReader stdin =
        new BufferedReader(new InputStreamReader(System.in));

    public static String rivi() {
        String arvo=null;
        boolean ok;
        do {
            try {
                arvo = stdin.readLine();
                ok = true;
            } catch (Exception e) {
                System.out.println("Virhe rivin lukemisessa. Anna uusi!");
                ok = false;
            }
        }
        while (!ok);
        return arvo;
    }
/*****/
    public static int kluku() {
        int arvo=-1;
        boolean ok;
        do {
            try {
                arvo = Integer.parseInt(stdin.readLine());
                ok = true;
            } catch (Exception e) {
                System.out.println("Kelvoton luku. Anna uusi! (virhe 02)");
                System.out.print("> ");
                ok = false;
            }
        }
        while (!ok);
        return arvo;
    }
/*****/
    public static double dluku() {
        double arvo=-1;
        boolean ok;
        do {
            try {
                arvo = new Double(stdin.readLine()).doubleValue();
                ok = true;
            } catch (Exception e) {
                System.out.println("Kelvoton desimaaliluku. Anna uusi!");
                ok = false;
            }
        }
        while (!ok);
    }
}
```

```
        return arvo;
    }

    /**
     *
     */
    public static long lluku() {
        long arvo=-1;
        boolean ok;
        do {
            try {
                arvo = new Long(stdin.readLine()).longValue();
                ok = true;
                if (arvo < 0) {
                    System.out.println("Kelvoton luku. Anna uusi! (virhe 03)");
                    System.out.print("> ");
                    ok = false;
                }
            } catch (Exception e) {
                System.out.println("Kelvoton luku. Anna uusi! (virhe 03)");
                System.out.print("> ");
                ok = false;
            }
        } while (!ok);
        return arvo;
    }

    /**
     *
     */
    public static char merkki() {
        String rivi = rivi();
        try {
            return rivi.charAt(0);
        } catch (Exception e) {
            return ' ';
        }
    }

    /**
     *
     */
    public static byte[] tavutTiedostosta(File file) {
        byte[] bytes = null;

        try {
            InputStream is = new FileInputStream(file);

            // Get the size of the file
            long length = file.length();

            // Create the byte array to hold the data
            bytes = new byte[(int)length];

            // Read in the bytes
            int offset = 0;
            int numRead = 0;
            while (offset < bytes.length
                && (numRead=is.read(bytes, offset, bytes.length-offset)) >= 0)
            {
                offset += numRead;
            }

            // Ensure all the bytes have been read in
            if (offset < bytes.length) {
                throw new IOException("Could not completely read file " +
file.getName());
            }

            // Close the input stream and return bytes
            is.close();
        }
    }
}
```

```
    } catch (Exception e) {
        System.out.println("Virhe tiedoston tavujen lukemisessa");
        return null;
    }

    return bytes;
}

/*****
public static String riviPuskurista(BufferedReader r) {
    String str = null;

    try {
        do {
            str = r.readLine();
            if(str != null) {
                return str;
            }
        } while (str != null);

        return null;
    } catch(Exception e) {
        System.out.println("Error:" + e );
        return null;
    }
}
*****/
```