

```

package correos;
import java.io.*;
import java.util.ArrayList;

/**
 * Clase con utilidades para la entrada de datos desde teclado y fichero
 * @author jvalvarez
 */
public class MyInput {
    // Lee una cadena de caracteres desde el teclado

    /**
     * Método que permite leer una cadena de caracteres del teclado
     * @return retorna una cadena de caracteres
     */
    public static String readString() {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in),1);
        String string="";
        try {
            string = br.readLine(); }
        catch (IOException ex) {
            System.out.println(ex); }
        return string; }
    // Lee un dato tipo int desde el teclado

    /**
     * Método que permite leer un número entero de simple precisión por teclado
     * @return retorna un número entero de precisión simple
     */
    public static int readInt(){
        return Integer.parseInt(readString()); }
    // Lee un dato tipo double desde el teclado

    /**
     * Método que permite leer número real por teclado.
     * @return retorna un número de doble precisión.
     */
    public static double readDouble() {
        return Double.parseDouble(readString()); }

    /**
     * Método que permite leer un número entero por teclado.
     * @return retorna un número entero comprendido entre -128 y 127
     */
    public static byte readByte() {
        return Byte.parseByte(readString()); }
    // Lee un dato tipo short desde el teclado

    /**
     * Método que permite leer un número entero por teclado.
     * @return retorna un número entero comprendido entre -32768 y 32767
     */
    public static short readShort() {
        return Short.parseShort(readString()); }
    // Lee un dato tipo long desde el teclado

    /**
     * Método que permite leer un número entero de doble precisión por teclado
     * @return retorna un número entero de doble precisión.
     */
    public static long readLong() {
        return Long.parseLong(readString()); }

    //Lee un dato tipo float desde el teclado

    /**

```

```

        * Método que permite leer número real por teclado
        * @return retorna un número de precisión simple
        */
public static float readFloat() {
    return Float.parseFloat(readString()); }

/**
 *
 * @param nombreFichero Es un String que contiene el path del fichero de texto.
 * si el fichero está en el raíz del proyecto, este String coincide con el nombre
 * del fichero con su extensión.
 * @return Un ArrayList de String con todas las palabras contenidas en el fichero de
texto.
*/
public static ArrayList <String> leeFichero(String nombreFichero){
    ArrayList <String> v= new ArrayList <String>();
    File fichero=null;FileReader fr=null;
    BufferedReader br=null;
    try{
        fichero=new File(nombreFichero);
        fr=new FileReader(fichero);br=new BufferedReader(fr);
        String linea;
        while ((linea=br.readLine())!=null){
            v.add(linea);}
    }catch (Exception e){
        e.printStackTrace();}
    finally {
        try {if (null!= fr){fr.close();
            br.close();}
        }catch (Exception e1){
            e1.printStackTrace();}
    }return v;
}

public static <A> void serialize(A a, String nombreFichero) {
    System.out.println("Serializando...");
    try {
        FileOutputStream fos = new FileOutputStream(nombreFichero) ;
        ObjectOutputStream oos = new ObjectOutputStream(fos) ;
        oos.writeObject(a) ;
        oos.close();//añadido personal
    } catch (Exception e) {
        System.err.println("Problem: "+e) ;
    }
}
}
}

```