

```

import java.io.*;
import java.util.Calendar;
import java.util.Date;
import java.sql.Timestamp;
import java.text.SimpleDateFormat;

import javax.servlet.http.HttpSession;

////////////////////////////////////
//                                     //
//                                     ALQUILER                                     //
//                                     //
////////////////////////////////////
public class GestionLogs {
    //public static String caminoFich="/tmp/"+;

    public static void escribirLog(HttpSession sesion, String texto) {
        String caminoFich="/tmp/"+nomFichero();

        String login = null;
        SimpleDateFormat formato = new SimpleDateFormat("yyyyMMdd_HHmm");
        String instante = formato.format(new Date());

        FileWriter fichero = null;
        PrintWriter pw = null;
        try {
            fichero = new FileWriter(caminoFich,true);
            pw = new PrintWriter(fichero);

            // Se obtiene la clase y método invocante. Ver:
            // http://www.javalobby.org/java/forums/t67019.html
            Throwable t = new Throwable();
            StackTraceElement[] elements = t.getStackTrace();
            String invocante = elements[1].getClassName() + "." + elements[1].getMethodName();
            // Se obtiene qué usuario estaba autenticado
            if (sesion!=null) login = "["+sesion.getId()+"] "+(String) sesion.getAttribute("usuarioBC");
            if (login ==null)
                if (sesion!=null)
                    login = "["+sesion.getId()+"] (desconocido) " ;
                else
                    login = "[ ----> SIN SESION <---- ] (desconocido) " ;
            // Escribe la línea con el formato:
            // Fecha y hora [TAB] método invocante [TAB] usuario autenticado
            [TAB] texto incidencia
            pw.println(instante + "\t" + invocante + "\t" + login + "\t" + texto);
        } catch (IOException e) {
            GestionLogs.escribirLog(null, "EXCEPCIÓN: " + e.toString());
            e.printStackTrace();
        } finally {
            try {
                // nos asegurarnos que se cierra el fichero.
                if (null != fichero)
                    fichero.close();
            } catch (Exception e2) {
                GestionLogs.escribirLog(null, "EXCEPCIÓN: " + e2.toString());
                e2.printStackTrace();
            }
        }
    }
}

```

```

    }
}

private static String nomFichero(){
    java.util.Date today = new java.util.Date();
    Timestamp ahora= new java.sql.Timestamp(today.getTime());

    Calendar hoy=Calendar.getInstance(); // el del objeto actual (.getyear()
esta deprecated)
    hoy.setTime(ahora);
    String res="RegistroActividad_"+String.valueOf(hoy.get(Calendar.YEAR))+ "-"
"+String.valueOf(hoy.get(Calendar.MONTH))+ "-" +String.valueOf(hoy.get(Calendar.DA
Y_OF_MONTH))+ ".txt";

    ///+// Cadenas.alert("fichero : "+res);
    return (res);
}

// No hace falta usarlo. Es mucho más simple. Ver MostrarSucesosTrastienda
@Deprecated
public static String leerLog() {
    File archivo = null;
    FileReader fr = null;
    BufferedReader br = null;

    archivo = new File (nomFichero());
    String res="";
    try {
        // Apertura del fichero y creaci3n de BufferedReader para poder
        // hacer una lectura c3moda (disponer del m3todo readLine()).

        fr = new FileReader (archivo);
        br = new BufferedReader(fr);

        // Lectura del fichero
        String linea;
        while((linea=br.readLine())!=null)
            res+=linea;
    } catch(Exception e) {
        GestionLogs.escribirLog(null, "EXCEPCI3N: " + e.toString());
        e.printStackTrace();
    } finally {
        // En el finally cerramos el fichero, para asegurarnos que se
        // cierra tanto si todo va bien como si salta una excepci3n.
        try{
            if( null != fr ) {
                fr.close();
            }
        } catch (Exception e2) {
            GestionLogs.escribirLog(null, "EXCEPCI3N: " + e2.toString());
            e2.printStackTrace();
        }
    }
    return res;
}
}

```