



UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS



Laboratorio de Diseño Orientado a Objetos

Armando Rodrigo Botello Alanis

Grupo: 009

Matrícula: 1722884

Práctica de Laboratorio #9

Maestro: Miguel Salazar

Día: 05 de marzo de 2017

Contenido

Objetivo	3
Opinión	3
Preguntas de Reflexión	4

Objetivo

Al finalizar las actividades de este laboratorio, deberás ser capaz de comprender la finalidad de el patrón Singleton.

Pasos

Utilizando los archivos del laboratorio 7, y en total utilizando 2 clases, se haría lo siguiente.

En la clase que ya habíamos creado de ComentariosControlador.java, se le añadió:

```
Log L = Log.getInstance("C:\\Users\\....\\Log.txt"\\  
L.write("Exception")
```

Para así crear un archivo de texto dónde se registren todos los que entraron a la página que creamos en el laboratorio 7

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException, SQLException{  
    response.setContentType("text/html;charset=UTF-8");  
    Log L;  
    L = Log.getInstance("Log.txt");  
    L.write("message");  
  
    try (PrintWriter out = response.getWriter()) {  
        /* TODO output your page here. You may use following sample code. */  
        String accion = request.getParameter("accion");  
        String nombre = request.getParameter("nombre");  
        String comentario = request.getParameter("comentario");  
  
        if (accion.equals("comentario")){  
  
            try{  
                ComentariosDAO cdao = new ComentariosDAO();  
                ComentariosPOJO cpojo = new ComentariosPOJO();  
                cpojo.setNombre(nombre);  
                cpojo.setComentario(comentario);  
                cdao.insertar(cpojo);  
                response.sendRedirect("buscar.jsp");  
            }catch(IOException ex){  
                L.write("Exception");  
            }  
        }  
        if (accion.equals("buscar")){  
  
            try{  
                ComentariosDAO cdao = new ComentariosDAO();  
                ComentariosPOJO cpojo = new ComentariosPOJO();  
                cpojo.setNombre(nombre);
```

```

*
* @author arman
*/
public class Log {
    private final String fileName;
    private static Log instance;

    private Log(String fileName){
        this.fileName=fileName;
    }

    public static Log getInstance(String fileName){
        if (fileName==null){
            return new Log(fileName);
        }
        return instance;
    }

    private void write(String message){

        try (BufferedWriter br = new BufferedWriter(new FileWriter(fileName, true))) {
            DateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
            Calendar cal = Calendar.getInstance();

            //Create the name of the file from the path and current time
            String data = "\n" + dateFormat.format(cal.getTime()) + ": " + message ;
            br.write(data);
        }
        catch(Exception ex){

        }
    }
}

```

Y después se crea una clase llamada Log, que sería el singleton, donde se utilizará para tomar la fecha registrarla en el archivo de texto que se crea en ComentariosControlador

Preguntas de Reflexión

- ¿Qué ventajas identificas con el uso de un sistema de Logging de eventos?

Que se registra todos los movimientos que los usuarios van realizando

- ¿Qué ventajas tienes al utilizar una clase singleton?

Que como encapsula su única instancia, puede controlar de cuando y como acceden los usuarios

- ¿Qué "pros" y "contras" identificas al utilizar singleton vs clases estáticas?

El singleton puede implementar interfaces, así que se puede tratar como si solo fuera otra implementación