

**Ingeniería de Software**

**Tarea – Segundo Bimestre**

|  |  |
| --- | --- |
| Estudiante: | José Luis Caicedo Díaz |
| **Nro. de cédula:** | 1712682739 |
| **Correo:** | [luisjoseec@gmail.com](mailto:luisjoseec@gmail.com) |
| **Profesor Tutor:** | Ing. Marco Patricio Abab |
| **Período** | Segundo Bimestre: Octubre 2018 – Febrero 2019 |

# NOTAS DE CAMBIO

Resumen de las actualizaciones:

|  |  |  |  |
| --- | --- | --- | --- |
| **Estado** | **Versión** | **Fecha (dd/mm/aaaa)** | **Comentario** |
| Emisión Inicial | 1.0 | 10/01/2019 | Emisión Inicial |
| Modificación | 1.1 | 11/01/2019 | 1. Modificación pantallas de resultado. 2. Especificación de mensajes de error de la aplicación 3. Cambio de mensaje al confirmar un registro guardado. 4. Conversión a MAYÚSCULAS todo carácter ingresado en los campos “Nombre”, “Apellido” que se mostrarán en el archivo “Registro.txt” |

1. **Título:**

Informe desarrollo Actividad Nro. 2: “Uso de herramientas de administración de la configuración”.

Aplicación que mediante un formulario realiza el registro de personas y el resultado guarda a un archivo txt.

1. **Objetivo:**

* Conocer la importancia del uso de herramientas de administración para procesos de desarrollo de aplicaciones de software.
* Elaborar un programa fuente en JAVA cuyos resultados sean almacenados en un archivo de texto.

1. Especificaciones Técnicas Generales de Desarrollo:

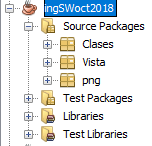
3.1 Descripción Técnicas de Desarrollo

Se creó el proyecto con el nombre: ingSWoct2018 utilizando los siguientes entornos de desarrollo:

|  |  |  |
| --- | --- | --- |
| **Nro.** | **Nombre** | **Versión** |
| **1.** | IDE Netbeens | Versión: 2018-09 (4.9.0)  Buid id: 20180917-1800 |
| **2.** | Java JDK | 1.8.0\_191 |
| **3.** | Rutime Java (TM) SE | 1.8.0\_191-b12 |
| **4.** | Lenguaje de Programación | JAVA |

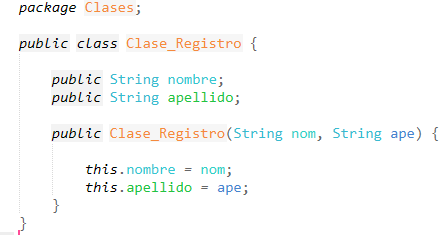
3.2 Estructura del desarrollo de la aplicación.

Para el desarrollo se ha creado dos paquetes de trabajo los cuales son el paquete “Clases” y el paquete “Vista”.



3.3 Implementación de los métodos que constituyen el programa desarrollado

Dentro de la clase: Clase\_Registro.java se encuentran creados los métodos a los cuales se va a instanciar desde la clase Registrar.java



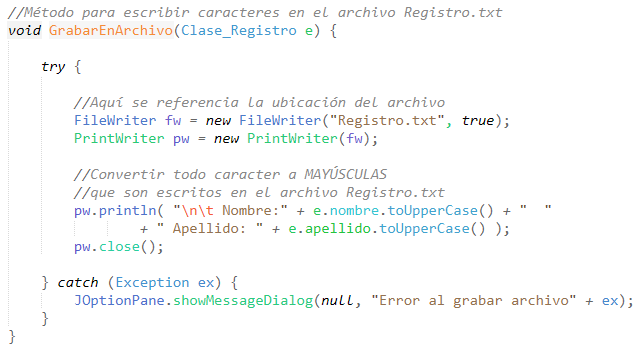
Mientras que, en la clase “Registrar.java” se construyen todos los métodos necesarios para realizar la creación del registro, escritura/lectura de datos y la conformación del formulario de presentación.

**Método GrabarEnArchivo()**

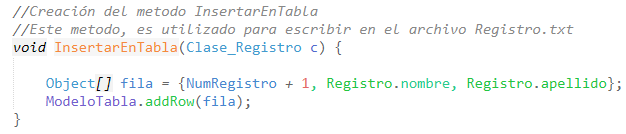
Este método escribir caracteres en el archivo texto **“Registro.txt”**, la ubicación del archivo en el cual se va a escribir el flujo de caracteres está especificada por la subclase **FileWriter.**

El uso de la subclase **FileWriter** nos obliga a capturar la Exception ex. Para cerrar la comunicación con el archivo **“Registro.txt”** se ha implementado el método **close().**

En el archivo **“Registro.txt”**, todo carácter ingresado lo convertirá a MAYÚSCULAS



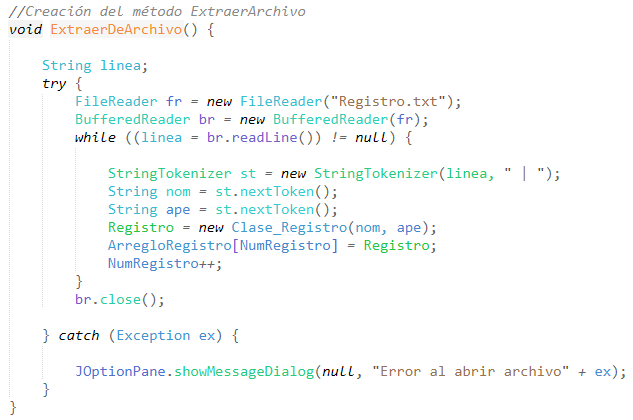
**Método InsertarEnTabla()**



**Método ExtraerDeArchivo()**

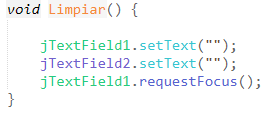
Se utiliza para realizar lecturas sencillas de caracteres que se encuentra dentro del archivo de texto “Registro.txt” La ubicación del archivo **“Registro.txt”** está especificada por la subclase FileReader.

El uso de la subclase **FileReader** nos obliga a capturar la Exception controlada. Para cerrar la comunicación con el archivo **“Registro.txt”** se ha implementado el método **close().**



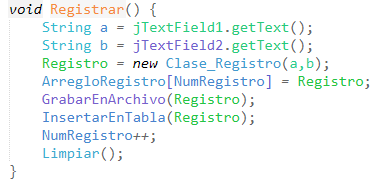
**Método Limpiar()**

Es encargado que luego de insertado los datos en la tabla, limpia los campos para ingresar el siguiente registro y el cursor retorna automáticamente al campo de nombre.

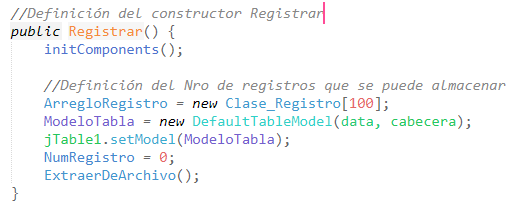


**Método Registrar()**

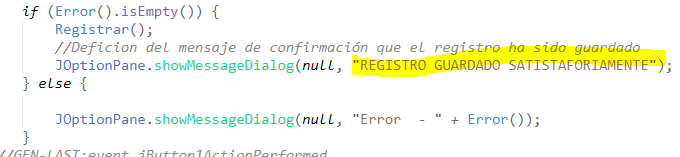
Registra en la tabla creada los datos ingresados en los campos pertenecientes a Nombre y Apellido



En esta sección de código se define la cantidad de caracteres que va recorrer el ArregloRegistro en este caso se ha programado que tenga una capacidad de 100 elementos dentro del arreglo.



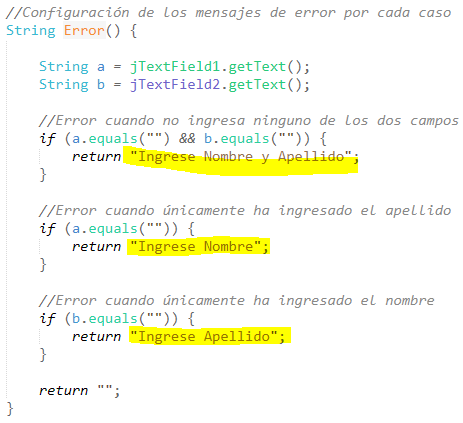
En esta sección de código se estructuran los mensajes de salida cuando un registro ha sido grabado correctamente y de la misma manera cuando se detectó un error durante la grabación.



**Método Error()**

Se utiliza este método para definir los mensajes de error que tiene la aplicación. Para este caso, se han definido tres tipos de errores:

1. Error al guardar un registro sin llenar los campos de “Nombre” y “Apellido”.
2. Error al guardar el registro únicamente ingresando datos en el campo “Nombre”.
3. Error al guardar el registro únicamente ingresando datos en el campo “Apellido”.

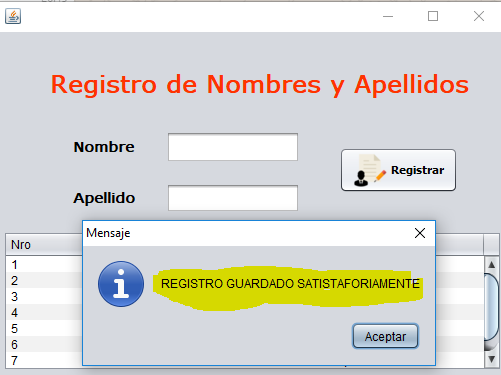


3.4 Presentación de Resultados

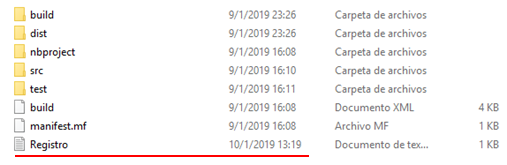
En la siguiente imagen se puede observar que al correr el programa salta el formulario en el cual se va ingresando los campos de “Nombre” y “Apellido”. Al presionar el botón “Registrar” se disparan las dos funcionalidades programadas, la primera guarda los datos ingresados en el archivo “Registro.txt” los cuales son mostrados dentro del jTable y la segunda produce limpia los campos de las etiquetas “Nombre” y “Apellido” para el siguiente registro que se desee ingresar



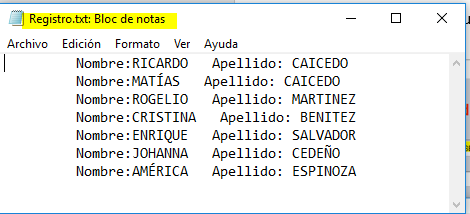
En la siguiente pantalla, muestra el mensaje “REGISTRO GUARDADO SATISFACTORIAMENTE” confirmando que el registro fue almacenado exitosamente.



En la siguiente imagen, se muestra la ubicación en la que se creó el archivo “Registro.txt”.



En el archivo “Registro.txt” se guardan los datos ingresados por medio del formulario en la cual todos los datos ingresados se realiza la conversión a letras “MAYÚSCULAS”.



Pantalla que muestran la configuración de los mensajes de error en los siguientes casos:

1. Error al guardar un registro sin llenar los campos de “Nombre” y “Apellido”.
2. Error al guardar el registro únicamente ingresando datos en el campo “Nombre”.
3. Error al guardar el registro únicamente ingresando datos en el campo “Apellido”.

