**ESCUELA POLITÉCNICA NACIONAL**

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

**ARQUITECTURA WEB**

|  |  |
| --- | --- |
|  | |
| ASIGNATURA: | BASE DE DATOS II |
| PROFESOR: | Ing. Juan Zaldumbide |
| PERÍODO ACADÉMICO: |  |
|  | |

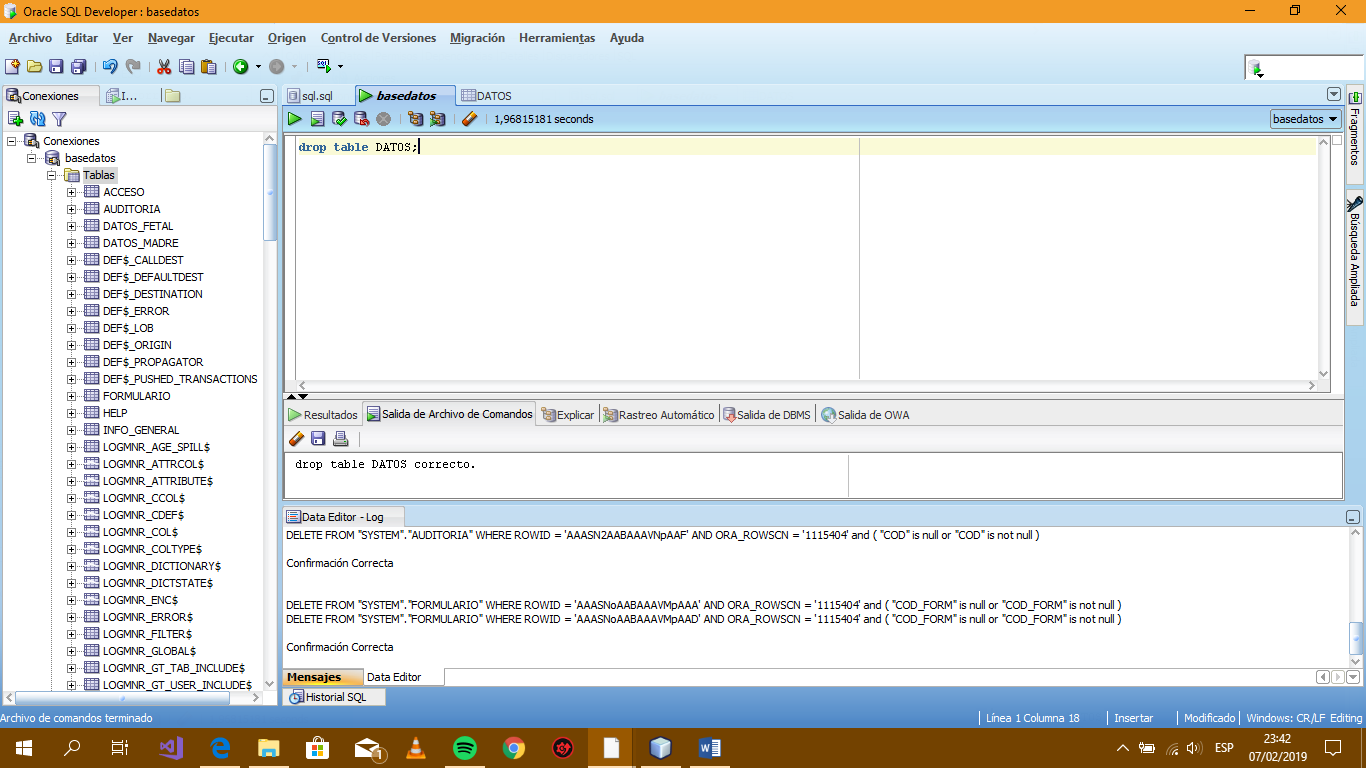
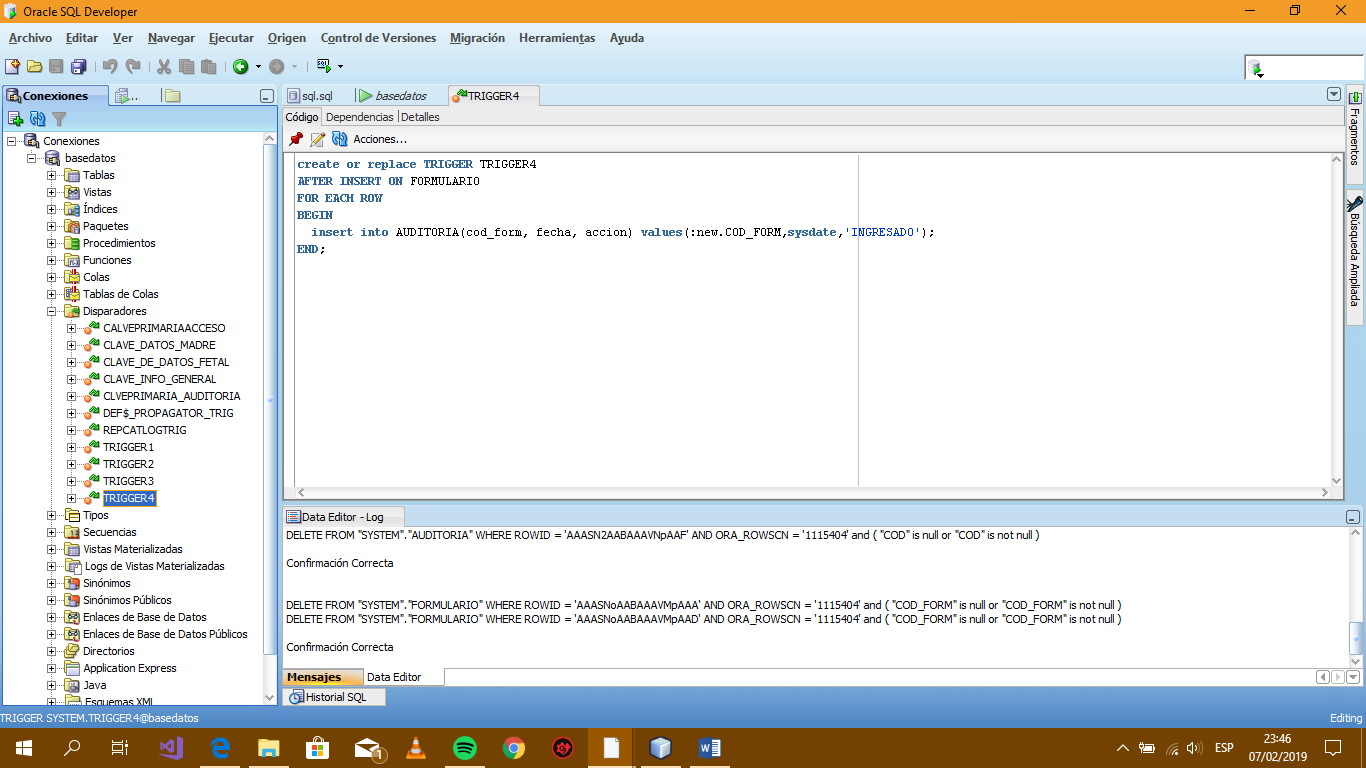
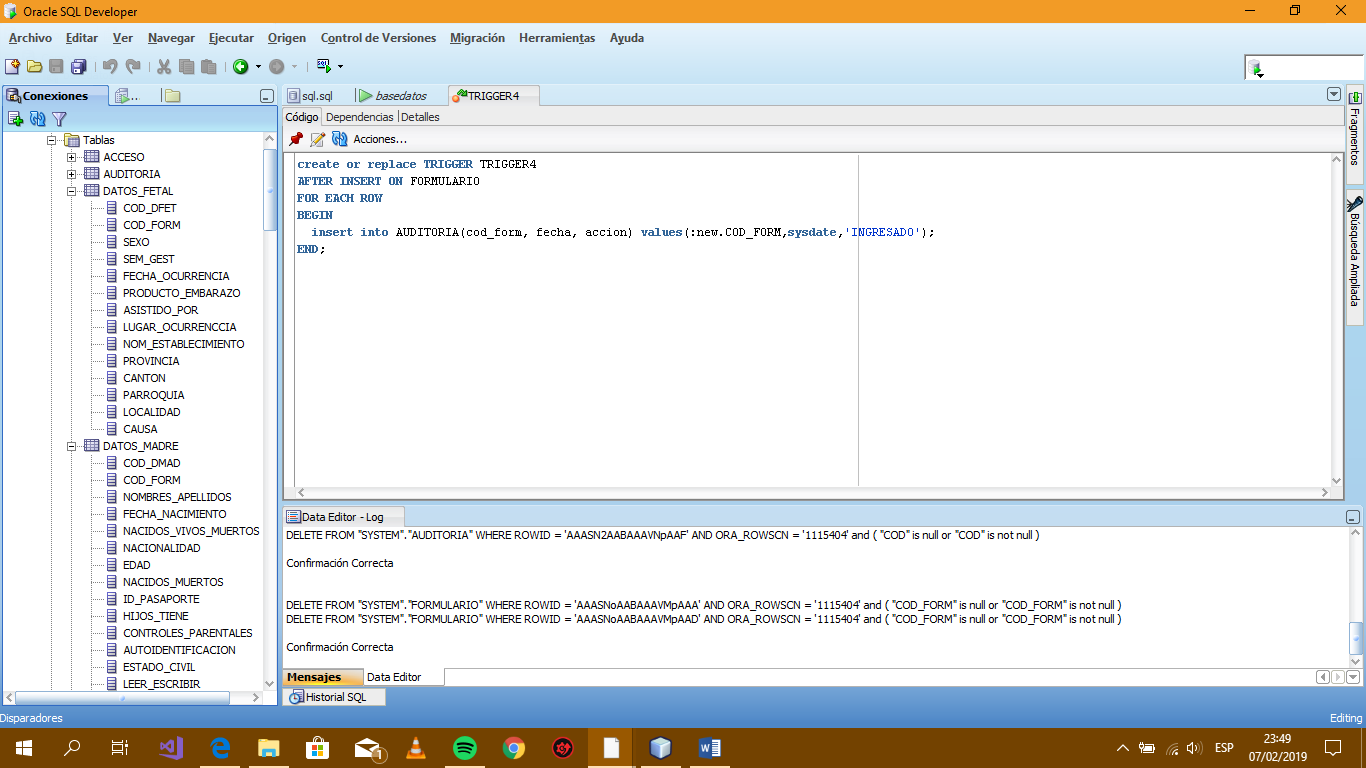
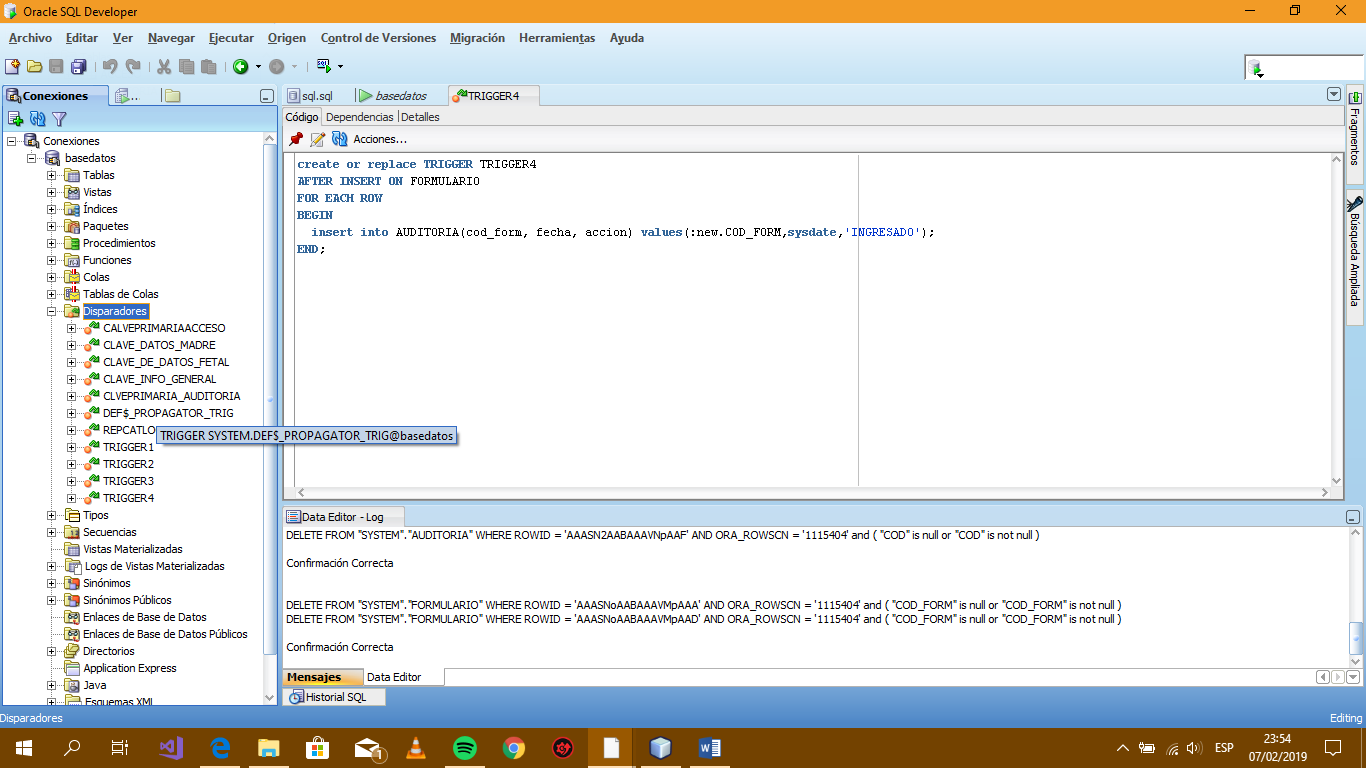
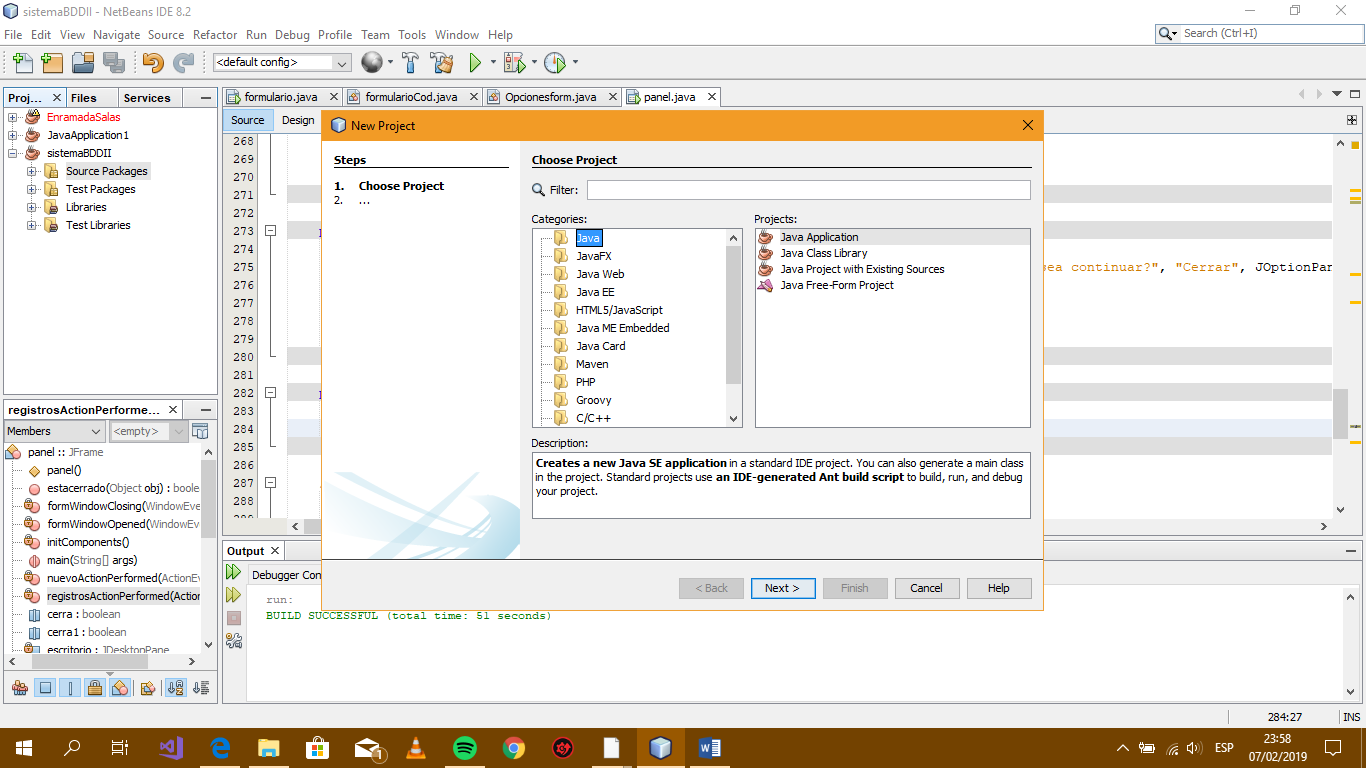
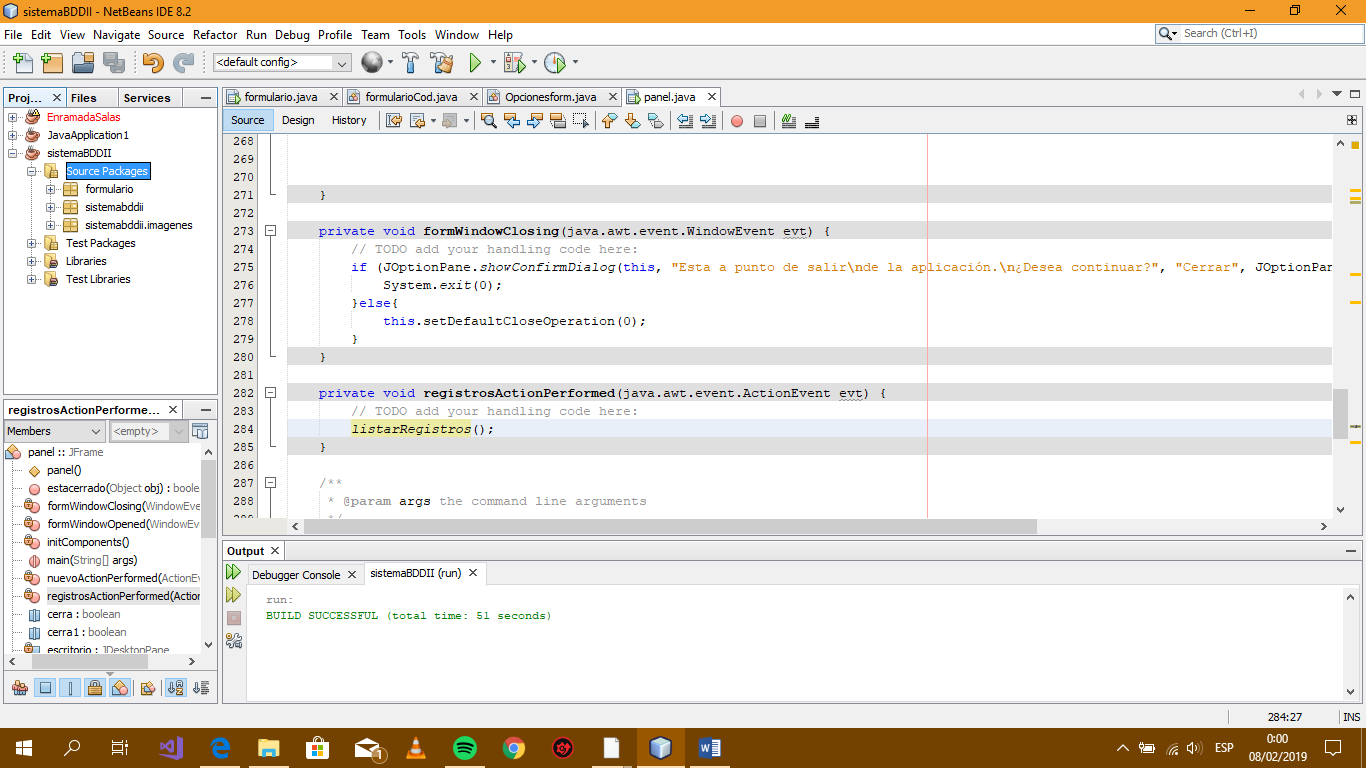
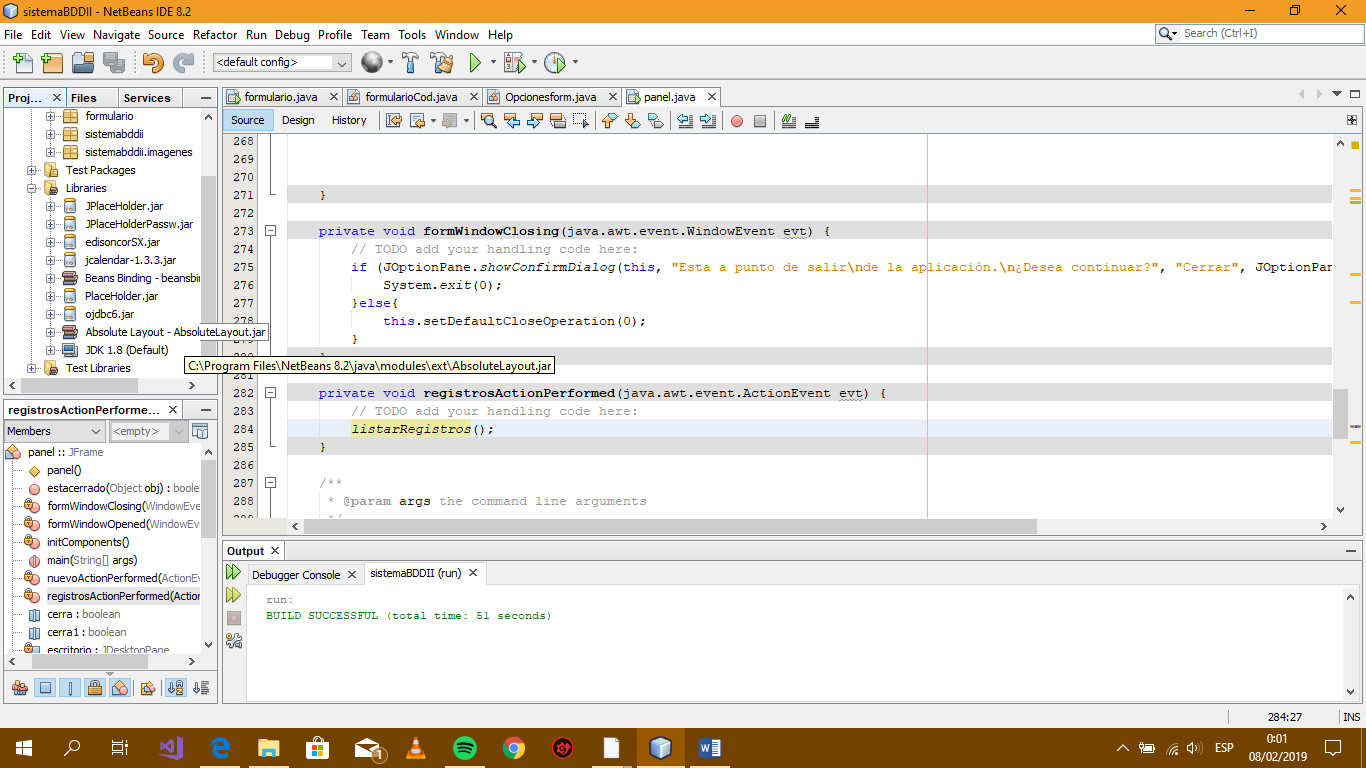
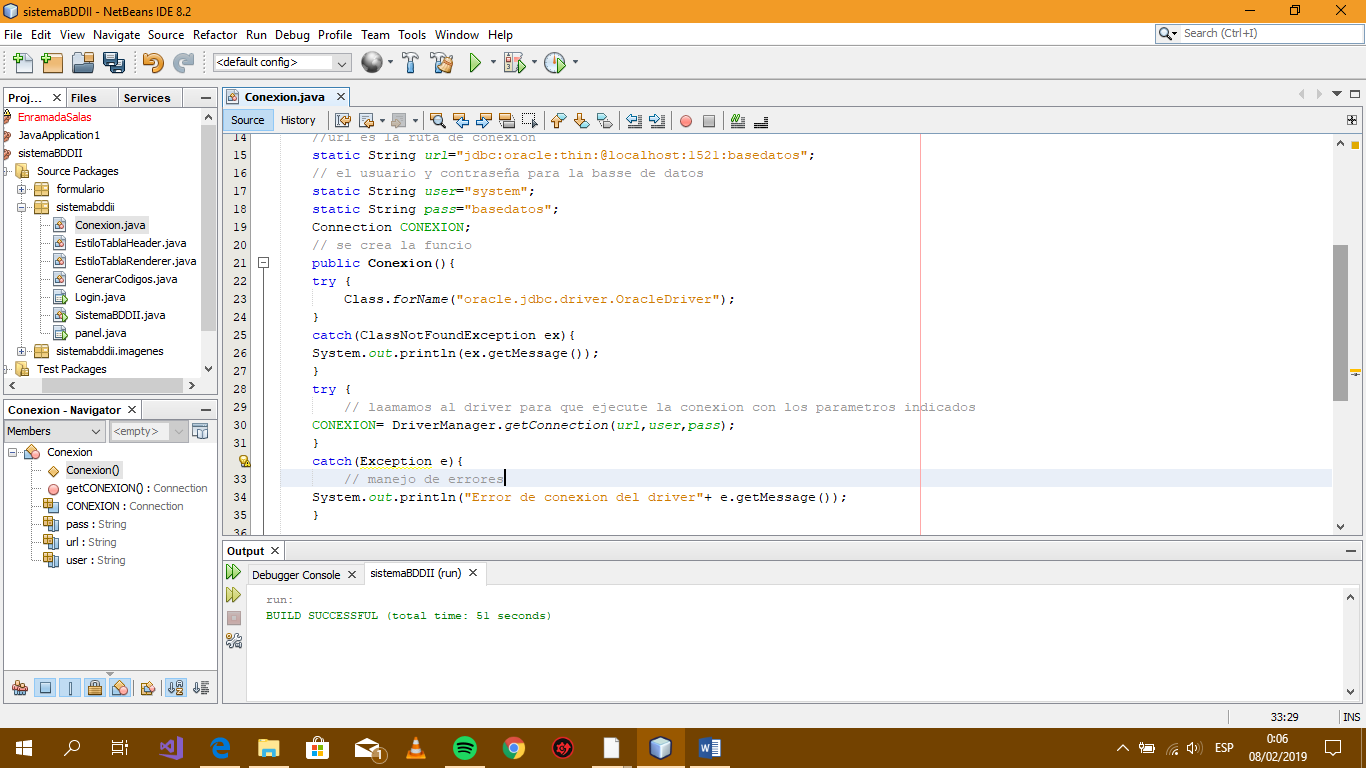
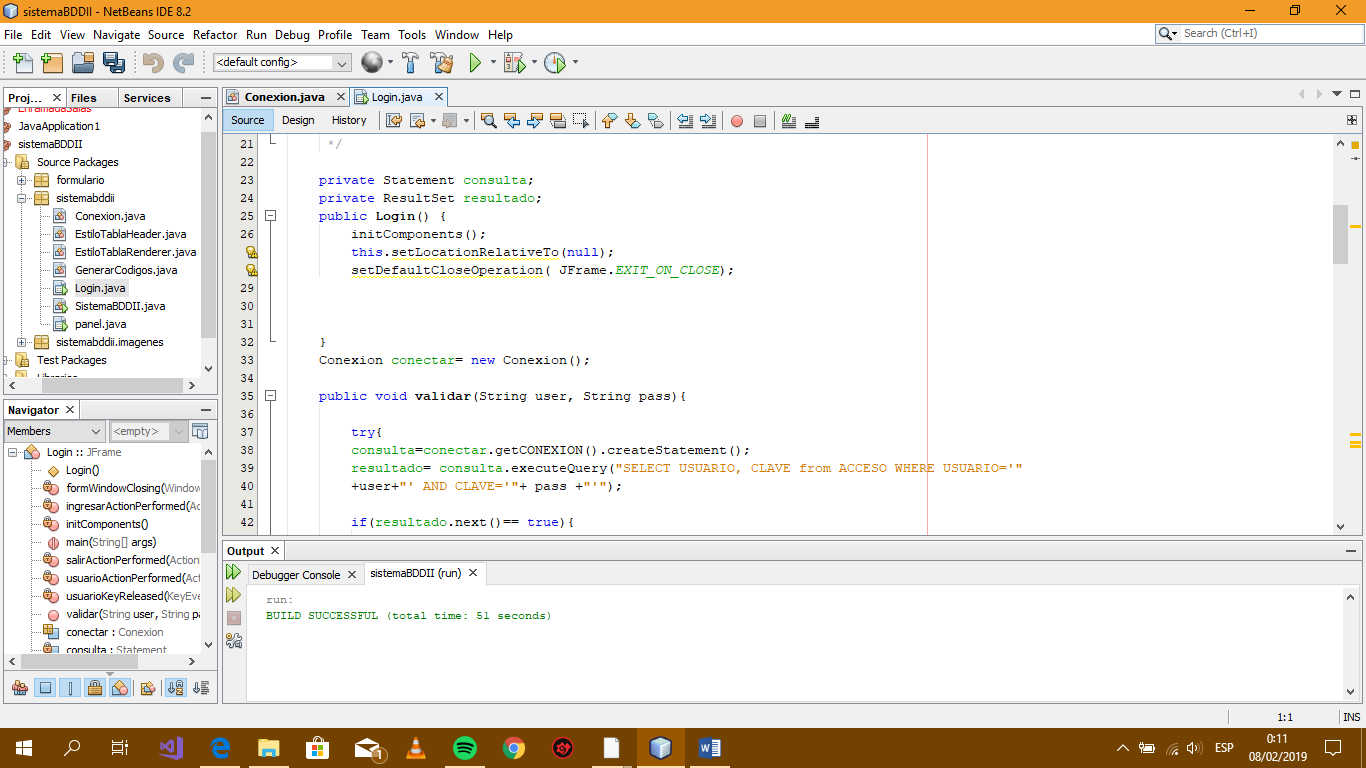
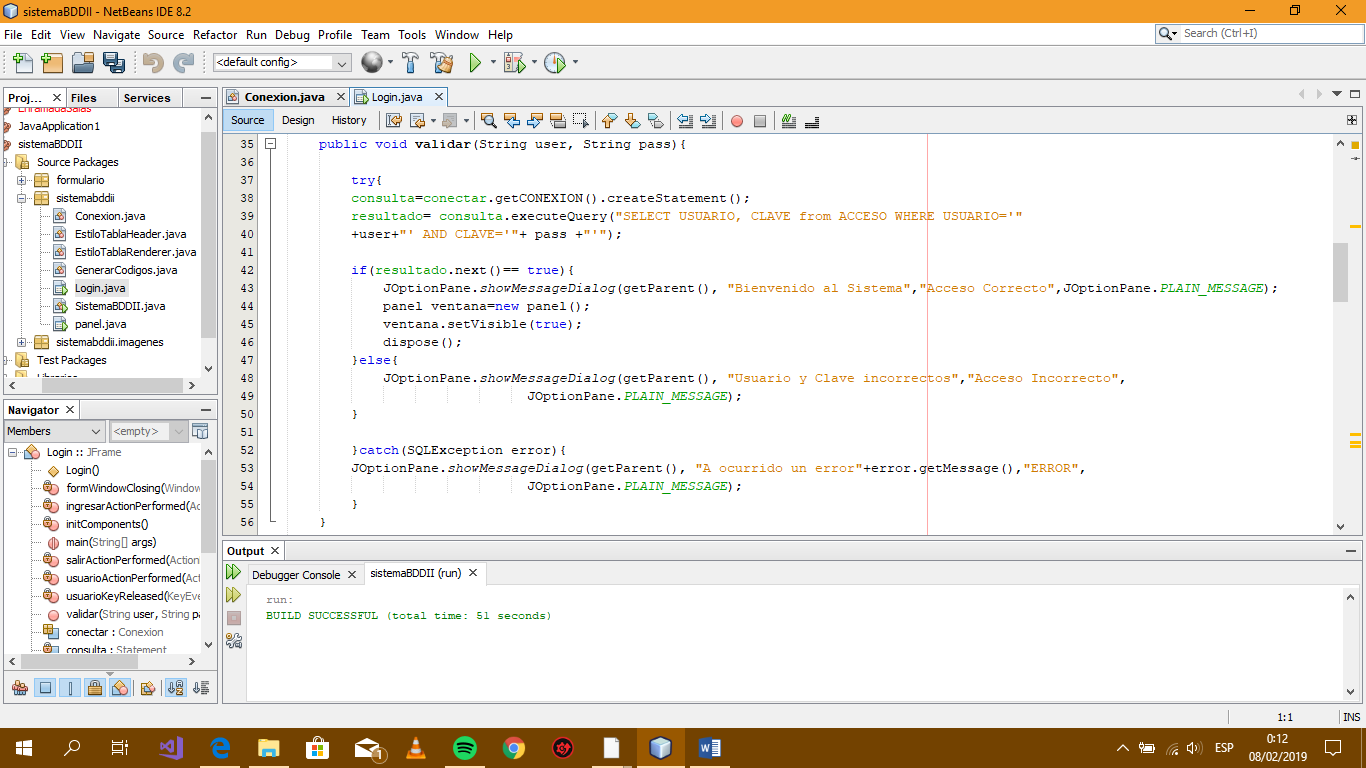
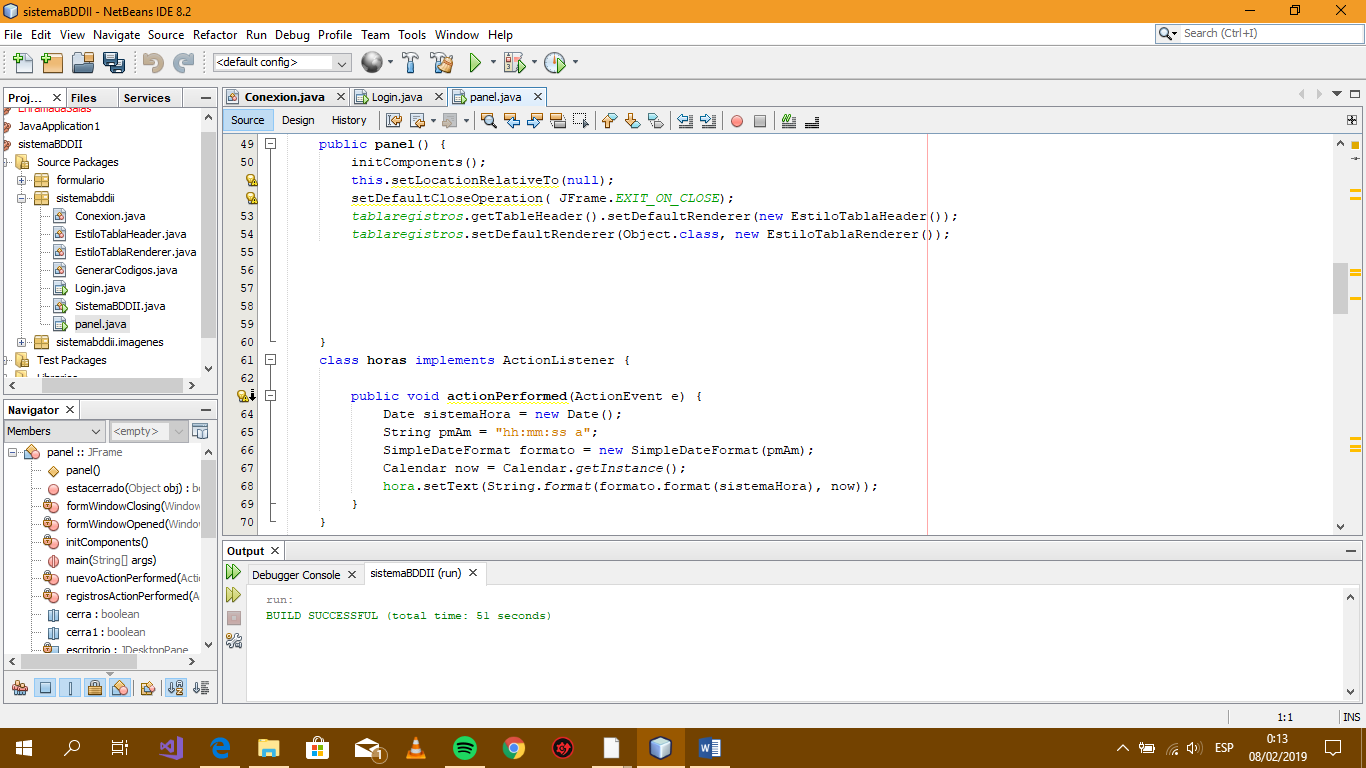
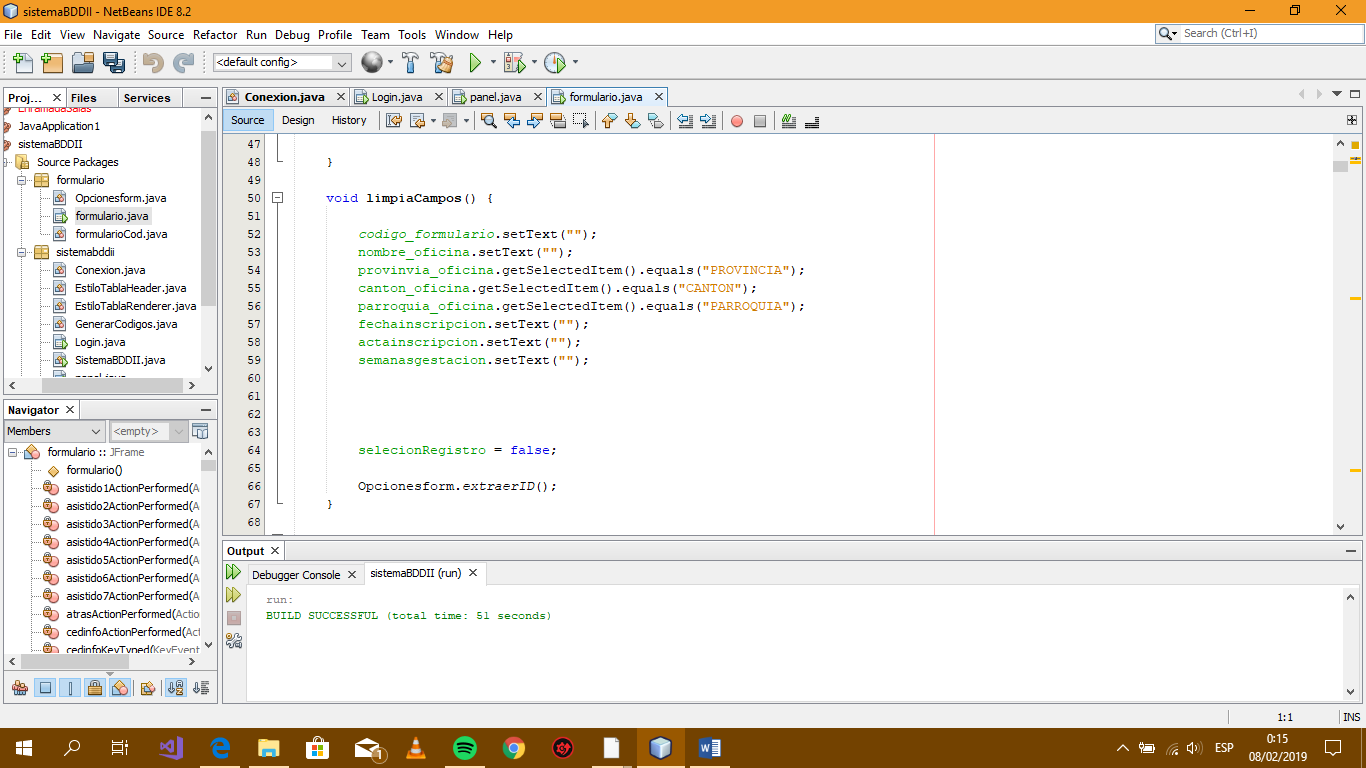
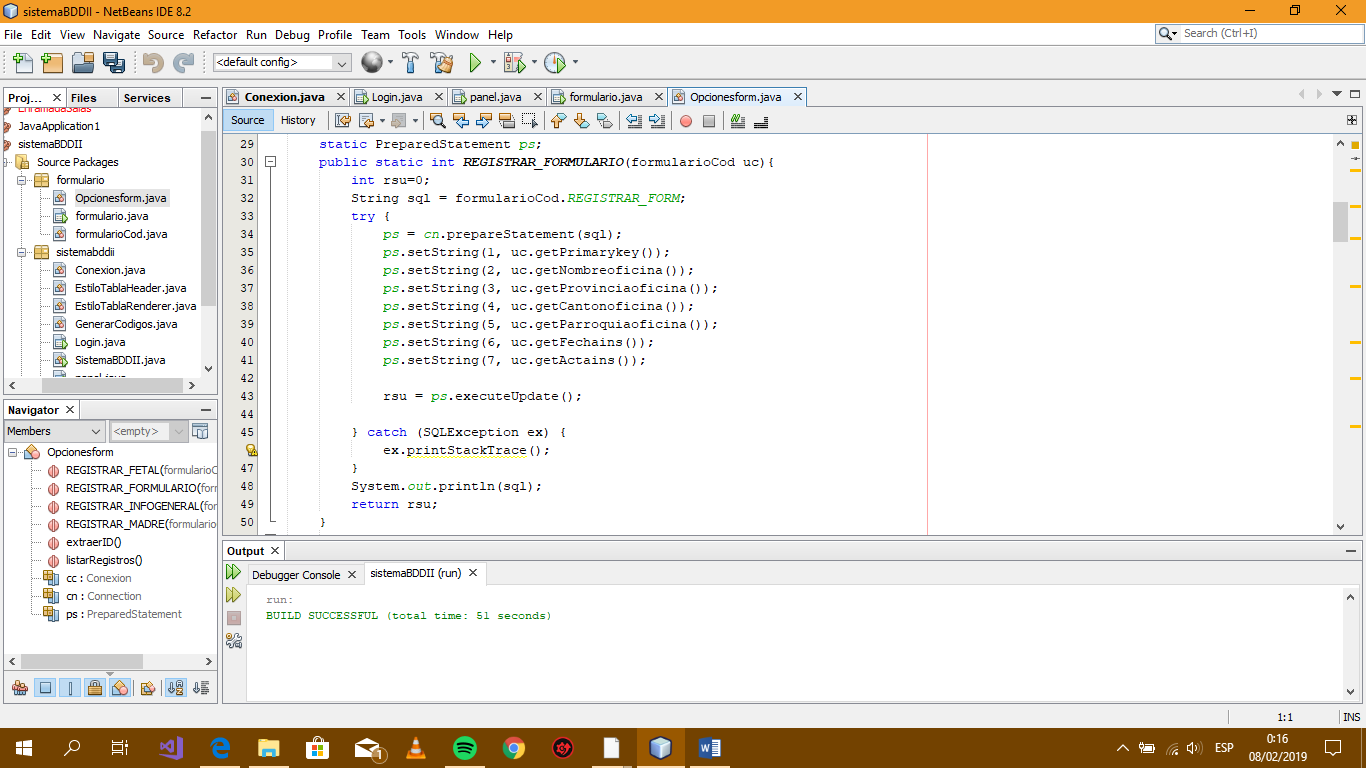
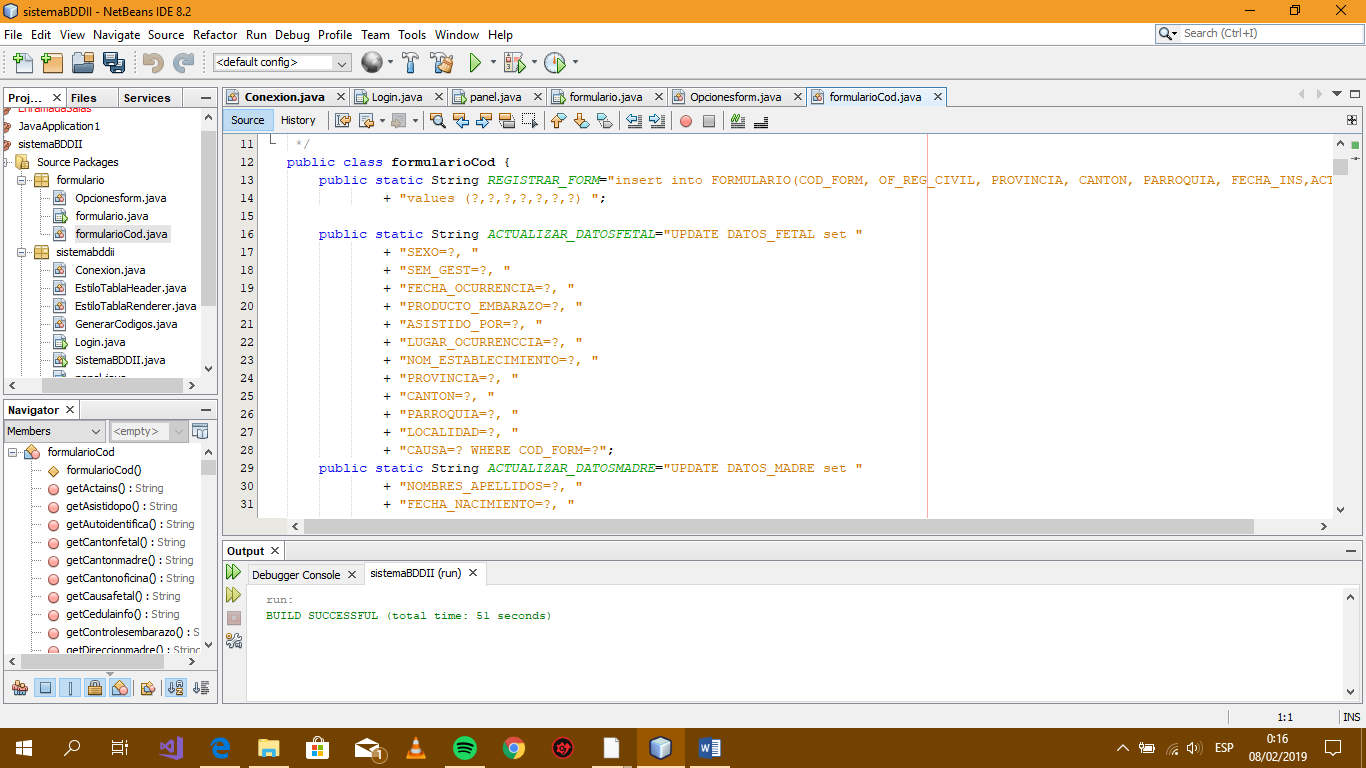
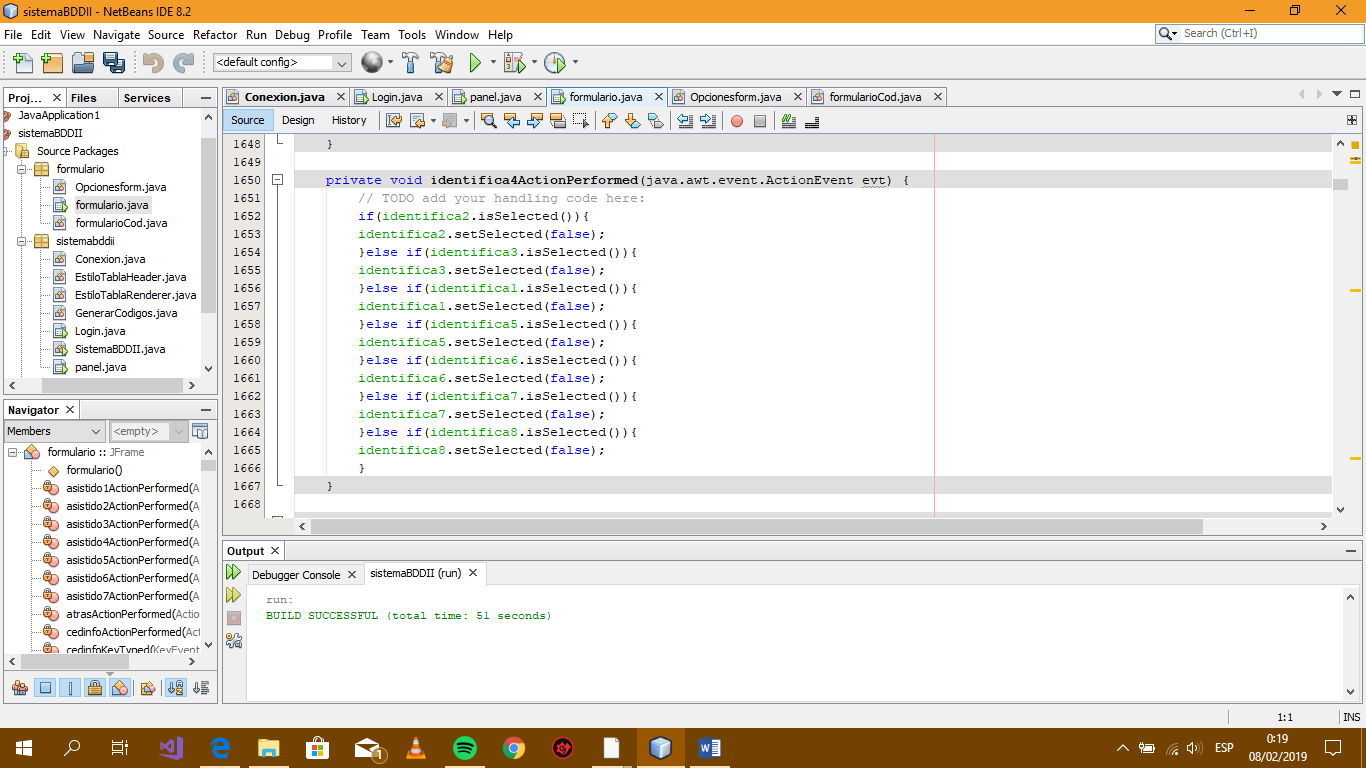
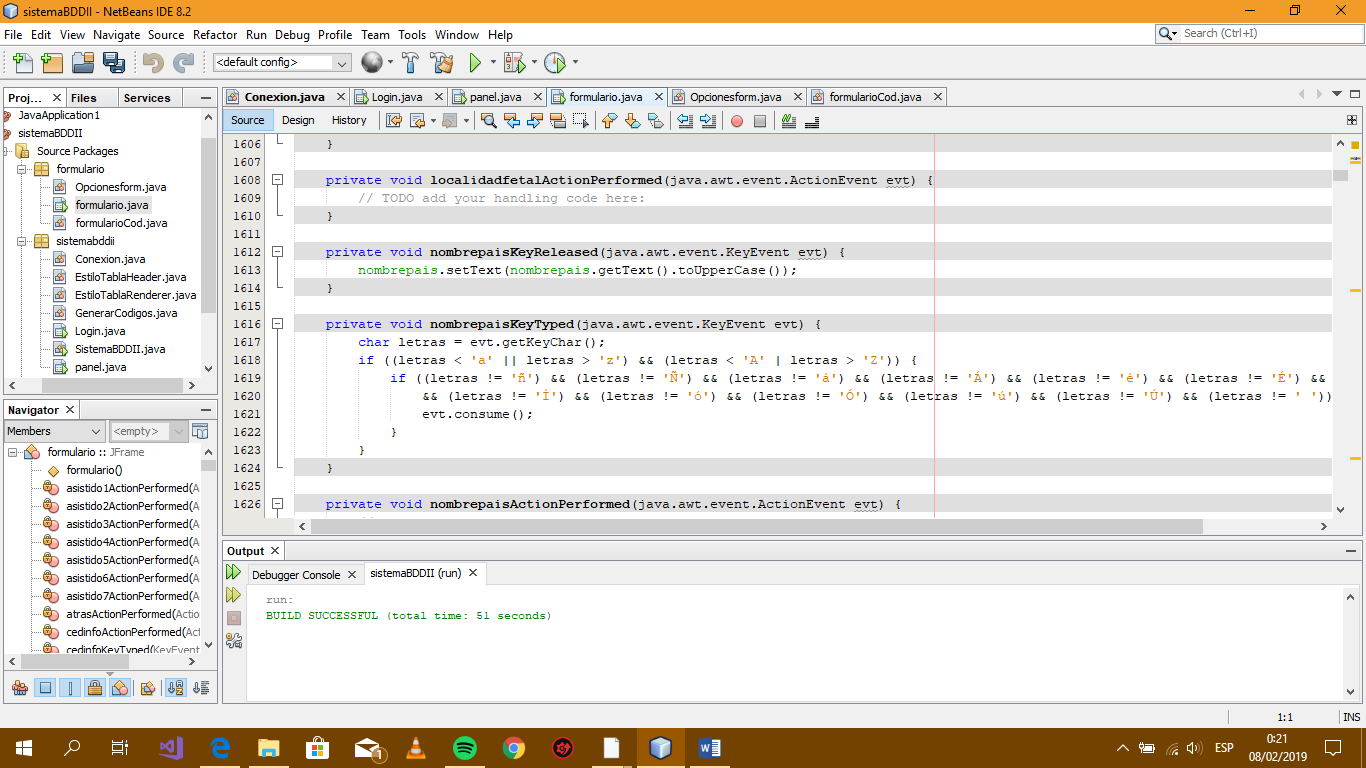
**APLICASION**

|  |  |
| --- | --- |
|  | |
| TÍTULO:  **Proyecto Final** | |
| **ESTUDIANTE** | |
| Carlos Farias | |
|  | |
|  | |
|  | |
|  | |
|  | |
| FECHA DE ENTREGA: | 07/ 02 / 2019 |
| CALIFICACIÓN OBTENIDA: |  |
| FIRMA DEL PROFESOR: |  |
|  | |

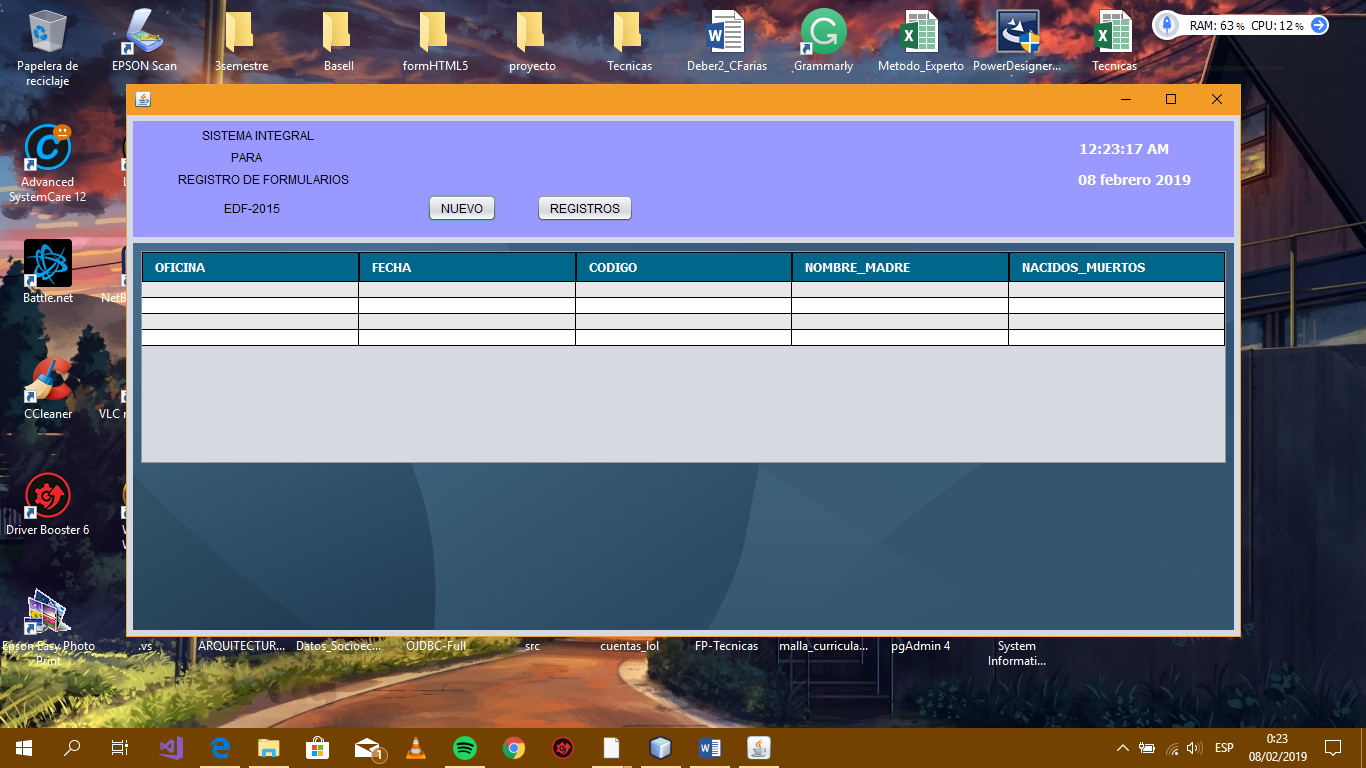
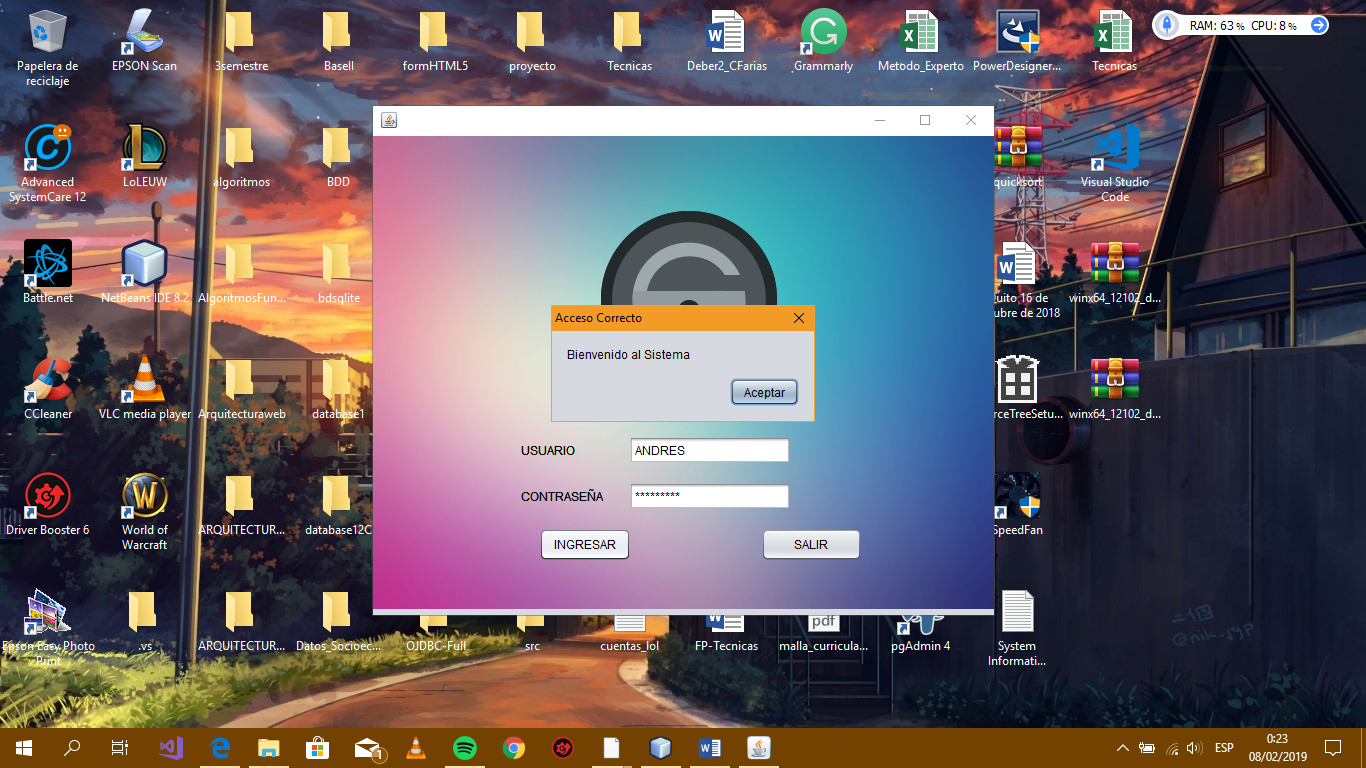
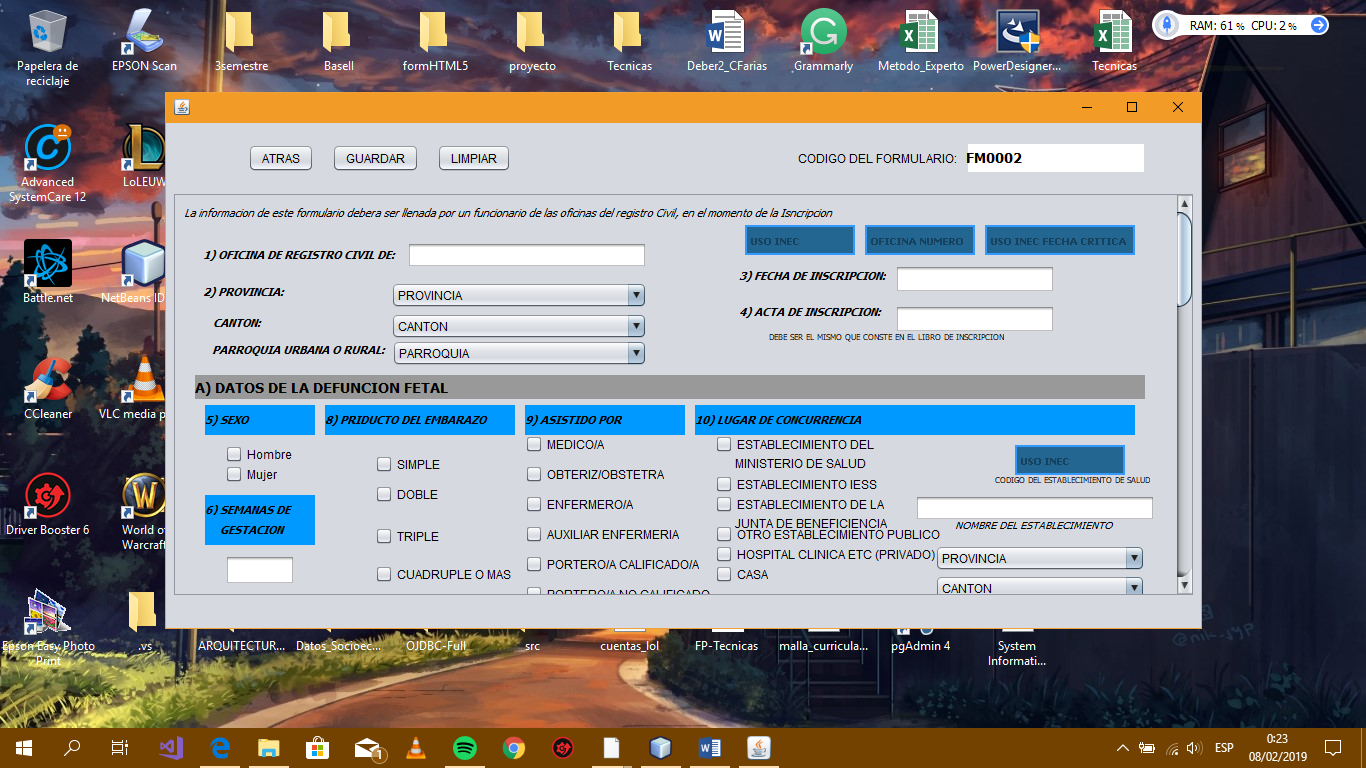
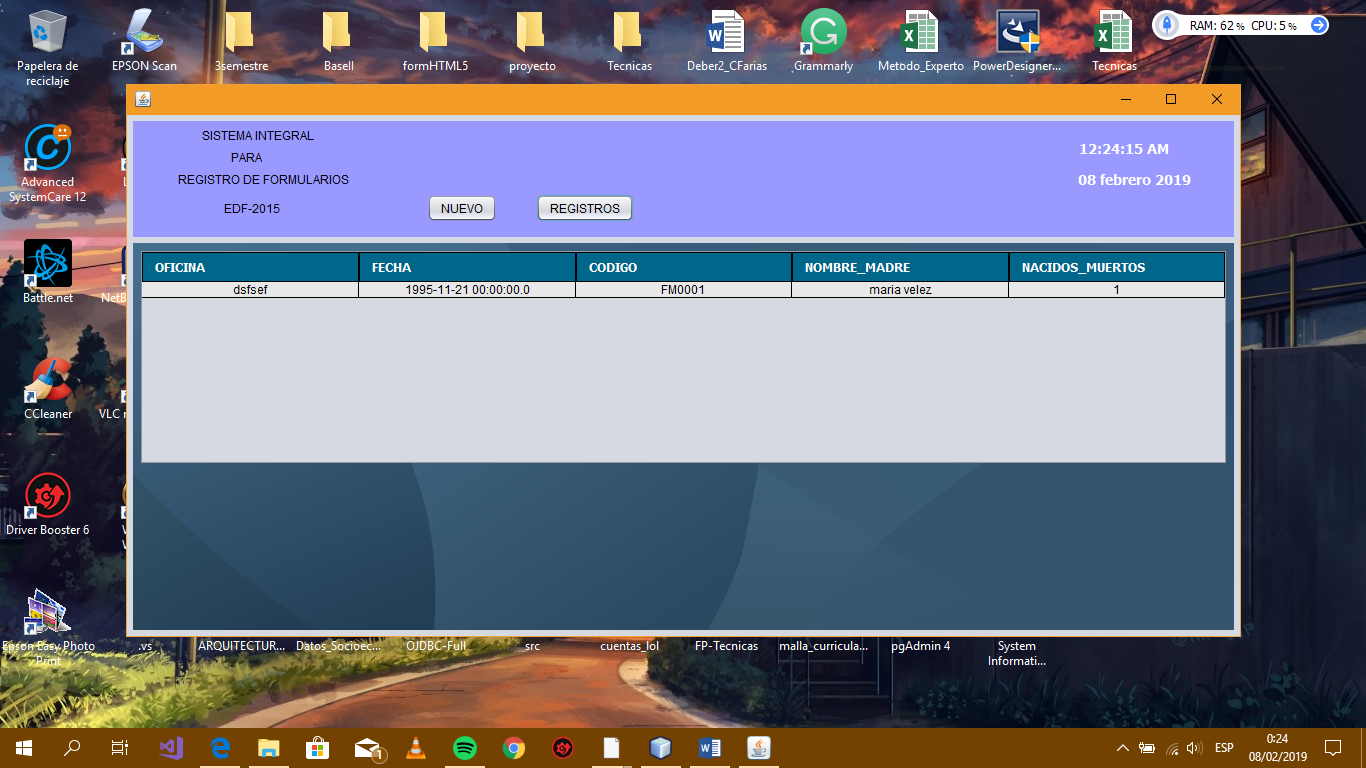
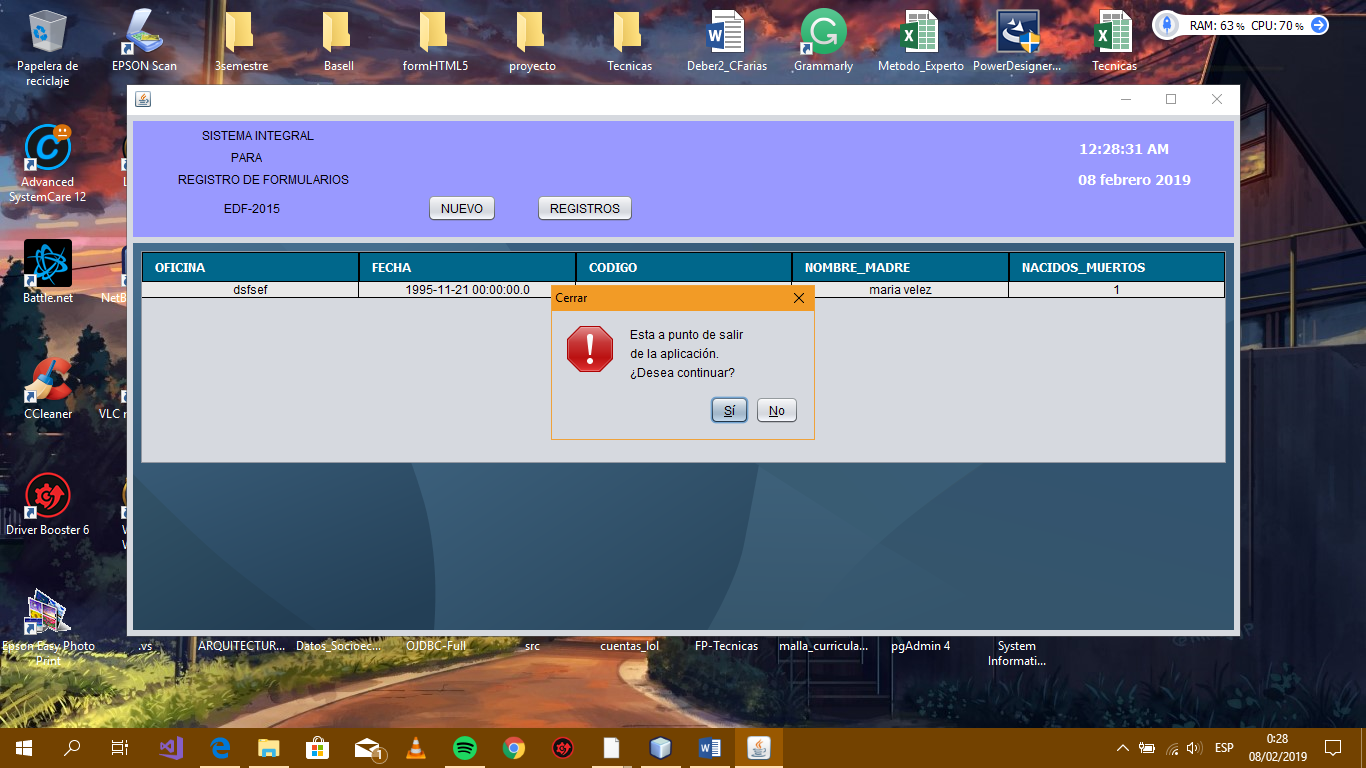
# OBJETIVOS ESPECÍFICOS

* Diseñar e implementar una base de datos para el formulario asignado
* Diseñar e implementar un sistema para ingresar información
* crear usuarios para control de acceso
* crear una tabla de auditoria

# DESARROLLO DE LA PRÁCTICA

* Se escogió como lenguaje de programación Java con su IDE NeatBeans por las facilidades que brinda
* El diseño de la base de datos fue hecho en Oracle developer
* 
* Se crearon 6 tablas en las que se van a almacenar los datos
* 
* Se dividió la información del formulario en cuatro tablas principales que fueron:
* Datos\_fetal que contenial la información del feto
* Datos\_madre que contenia la información de la madre
* Formulario que contiene la información del formulario
* Info\_general que contiene datos de información general del formulario
* También se creo una tabla de auditoria para saber en que momento se realizaron los registros
* Se creo una tabla de acceso que contiene los datos de usuarios que tienen acceso al sistema
* La tabla AUDITORIA se alimenta mediante un trigger de forma automática al momento de ingresar un registro este trigger llena ciertos campos de la tabla
* 
* Las tablas del formulario se relacionan mediante una clave principal que es COD\_FORM de la tabla Formulario y en el resto COD\_FORM FK para poder ingresar los registros de una forma más ágil
* 
* Se crearon disparadores para que el momento que se ingresa valores en la tabla FORMULARIO el código del formulario se guarde automáticamente en el resto de las tablas que corresponden al formulario
* Se crearon triggers para el proceso de autoincremento de claves de las distintas tablas
* 
* También se creo una vista para la aplicación
* Esta vista muestra los datos guardados de la tabla auditoria
* 
* El desarrollo de la aplicación fue en Neatbeans
* Se creo un proyecto llamado AplicacionBDDII
* 
* Dentro de nuestra aplicación se crearon tres paquetes
* Un paquete que es del formulario
* Otro paquete que es del main y la parte del login y panel principal
* Otro paquete que contiene imágenes
* 
* También se importaron alguna librerías entre las que destacan el jdk la versión de java proporcionada por neatbeans
* La librería OJDBC que es el controlador de conexión para Oracle
* El resto de librerias ayudan a mejorar la estética de nuestra aplicación
* 
* Para realizar la conexión se creo una clase especifica solo para eso llamada conexión
* 
* Definimos en la url que tiene la siguiente structura
* static String url="jdbc:oracle:thin:@localhost:1521:basedatos";
* tipo de variable – nombre- = ”driver:basededatos:tipo:servidor:puerto:SID”
* por trabajar con una conexión a base de datos siempre se maneja dentro de un try y un catch para manejar los errorer
* se creo una ventana de login
* 
* En esta ventana se va ha realizar el proceso de verificación de usuario y contraseña
* 
* También hay u panel principal que se mostrara al momento de tener éxito en el logeo al sistema
* 
* El formulario se lo creo dentro de su paquete debido a la cantidad de funciones que este requeria
* El JFRAME contiene el aspecto y funcionalidades de verificación, validación de datos
* 
* El archivo opcionesform contiene la funcionalidades como registrar datos y listar los datos
* 
* El archivo formulariocod contiene las sentencias utilizadas para insertar datos en la tabla de la base de datos
* 
* Incluyendo los getters and setter para obtener y definir los datos que van a ser insertados
* Debido a que le formulario incluye varias checkbos se procedió a realizar verificaciones de que solo uno este seleccionado y no mas de una a la vez
* 
* Para realizar la validación de campos se utilizaron los eventos de tipo key typed que nos permiten verificar que datos se están ingresando en los textfield
* 

# RESULTADOS

* 
* 
* 
* 
* 

# CONCLUSIONES

* La conexión a Oracle desde neatbeans no es muy complicada solo hay que saber cosas como el SID de la base de datos a la que queremos conectarnos a parte de la ip de servidor, el usuario y la contraseña correspondientes
* El diseño de la base de datos es muy importante en este tipo de casos ya que nos puede facilitar o complicar mucho las cosas al momento de realizar la inserción de datos por medio de la aplicación
* Los formularios son aplicaciones muy largas ya que debido a la cantidad de campos que tienen la validación de campos es muy larga de hacer.