Creamos una conexión a la base de datos Oracle y un usuario tal como indica el anunciado. Conexión TareaAD07 y usuario AD07 con la contraseña AD07.

Creamos una aplicación java “AppAlumnos”.

Modificamos la aplicación AlumnoBean para que se conecte a la base de datos Oracle.

# Class.forName("oracle.jdbc.driver.OracleDriver");

# Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl", "AD07", "AD07");

Añadimos un bloque try catch para manejar la excepción de null al recorrer el vector de Alumnos.

Depuramos la aplicación y la importamos a la biblioteca de la aplicación AppAlumnos.

Añadimos a la biblioteca el driver de conexión a la base de datos Oracle “ojdbc6.jar”

Creamos un panel personalizado.

Lo agregamos a la aplicación AppAlumnos importando él .jar a la biblioteca.

Creamos un JFrame que será la pantalla principal de la aplicación “Principal”

Arrastramos des de la paleta el Beans alumnoBeans y el Beans PanelPersonalizado.

Añadimos una imagen al panel personalizado desde sus propiedades.

Añadimos los botones de control de la aplicación.

Botón ver alumnos que nos abre un JDialog mostrando un atabla con los alumnos que hay en la base de datos.

El el JDialog verAlumnos añadimos el Beans alumnoBeans.

Creamos un botón añadir alumno que nos va a permitir añadir alumnos a la base de datos a través de un formulario.

El botón lanzara un jDialog con un formulario para rellenar los datos del alumno a añadir, incluye un campo fecha de nacimiento que nos envía a un jCalendar el cual hemos descargado de internet e importado a la paleta añadiéndolo a la biblioteca de proyecto.

En la pantalla “anadirAlumno” hay que implementar BDModificadaListene del beans alumnos para poder gestionar los cambios que se producirán en la base de datos cuando se añada un nuevo alumno “AlumnoBean.BDModificadaListene”

Añadimos un nuevo botón “Consultar alumno” que abrirá un nuevo dialogo con en formulario. A través del cual indicando el dni del alumno obtendremos los datos del alumno.

Cambiamos la función “seleccionarDNI” para que recorra todos los datos de la base de datos en busca de alguna coincidencia.

# public void seleccionarDNI(String nDNI) {

# Alumno a = new Alumno();

# this.DNI = "";

# this.Nombre = "";

# this.Apellidos = "";

# this.Direccion = "";

# for (int j = 0; j < Alumnos.size(); j++) {

# a = (Alumno) Alumnos.elementAt(j);

# System.out.println(a.DNI);

# if (a.DNI.equals(nDNI)) {

# this.DNI = a.DNI;

# this.Nombre = a.Nombre;

# this.Apellidos = a.Apellidos;

# this.Direccion = a.Direccion;

# this.FechaNac = a.FechaNac;

# }else{

# System.out.println("no hay dni igual");

# }

# }

Después de cambia la clase AlumnoBeans hay que copilar la clase y borrar el bean de la pantalla consultarAlumno y volverlo a añadir para que los cambios se hagan realidad.

Nota el archivo .jar que se importa ha de coincidir con el archivo .jar del proyecto en el que trabajamos con la clase para que estos cambios se hagan efectivos. Si se copia él .jar y se añade en alguna carpeta del proyecto hay que sustituirlo por el que hemos actualizado.