

**Tema:** Tu aplicación java + BD

En este tutorial vamos a crear una aplicación de escritorio simple y la vamos a conectar con una base de datos.

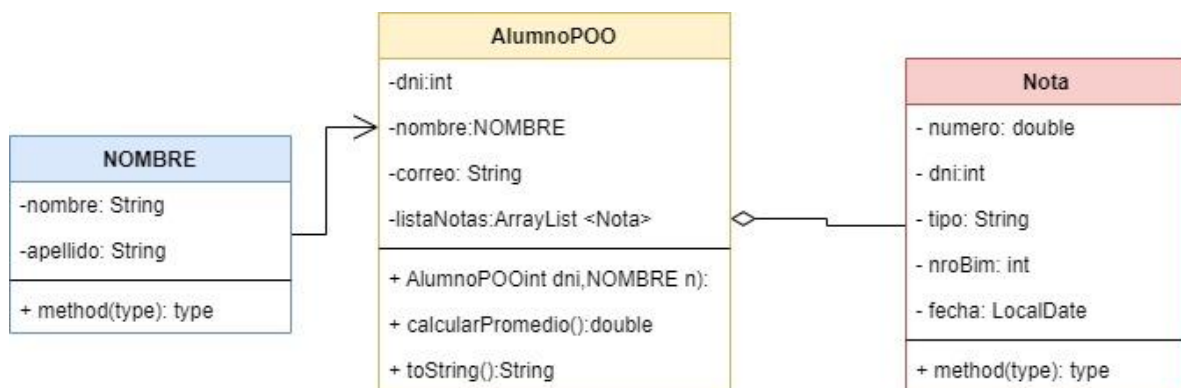
1. Ejecuta en la consola el siguiente script para crear nuestra BD “**notasQuinto**”, con las tablas “**alumnoPOO**” y “**notas**”

**SQL**

```

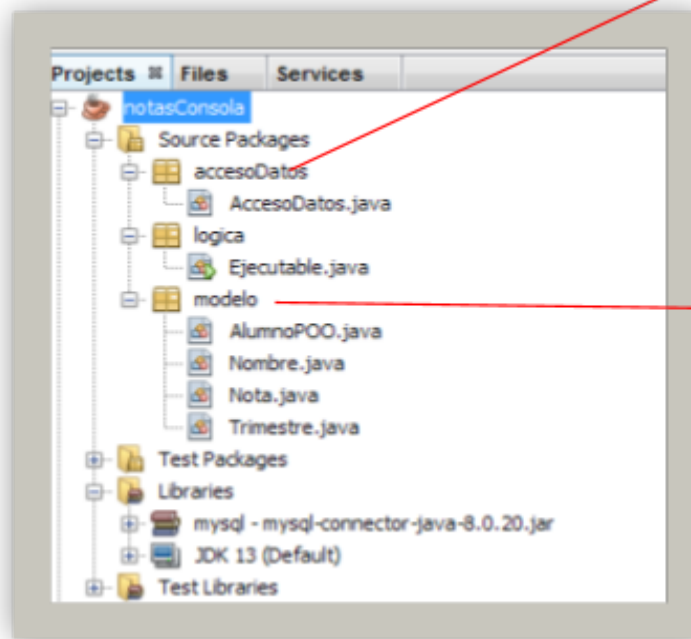
1  create database notasQuinto;
2  use notasQuinto;
3  create table alumnoPOO(
4      dni int,
5      nombre varchar(30),
6      apellido varchar(30),
7      email varchar (50) ,
8      primary key (dni)
9  );
10 create table nota (
11     id int auto_increment,
12     dni int,
13     trimestre int,
14     valor int,
15     motivo varchar(100),
16     primary key (id),
17     foreign key(dni) references alumnoPOO(dni)
18 );

```

**En Draw.io**

**En NetBeans**


2. Debemos lograr la siguiente estructura, donde



accesoDatos, será el package que aloje nuestra clase AccesoDatos.java; quien contiene todo el código necesario para operar con una Base de Datos

modelo, tendrá todas las clases que se describen en el DC.

**Agregamos la dependencia para conectar con mysql**

 com.mysql.cj.jdbc

**Clase AccesoDatos.java**

En esta clase será necesario importar todas las librerías para operar con BD y todas las clases que se encuentren en el package modelo

```

1  package accesoDatos;
2
3  import java.sql.Connection;
4  import java.sql.DriverManager;
5  import java.sql.ResultSet;
6  import java.sql.SQLException;
7  import java.sql.Statement;
8  import java.util.Vector;
9  import java.util.logging.Level;
10 import java.util.logging.Logger;
11 import modelo.AlumnoPOO;
12 import modelo.Nombre;
13 import modelo.Nota;
```

```

15 public class AccesoDatos {
16     private Connection conn;
17     private Statement comandos;
18
19     + public AccesoDatos() {...15 lines }
34     + public void insertarAlumno(AlumnoPOO unAl) {...11 lines }
45     + public ArrayList<AlumnoPOO> dameLista() {...17 lines }
62     + public ArrayList<ArrayList<String>> dameTabla() {...21 lines }
83     + public String dameDNI(String apellido) {...13 lines }
96     + public void insertarNota(Nota n) {...11 lines }
107 }

```

### Método constructor

Este método nos permite cargar el driver para la conectar con la BD.

```

18 public AccesoDatos() {
19     try {
20         Class.forName("com.mysql.cj.jdbc.Driver");
21         conn=DriverManager.getConnection("jdbc:mysql://localhost/notasQuinto",
22                                         "root", "");
23         /*en caso que:
24         1) No se puede conectar con el servidor de la BD,
25         2) usuario o pass incorrecto */
26         comandos=conn.createStatement();
27     } catch (ClassNotFoundException ex) {
28         System.out.println("Error:No se encontró la BD");
29     } catch (SQLException ex) {
30         System.out.println("Error:Usuario o pass incorrecto");
31     }
32 }

```

Instanciamos un objeto  
Connection

**Método insertarAlumno**

```
35 public void insertarAlumno(AlumnoPOO unAl) {
36     String sql="insert into alumnoPOO values("+unAl.getDni()+ ", '"+
37         unAl.getNomApe().getNombre()+", '"+
38         unAl.getNomApe().getApellido()+", '"+ unAl.getEmail() +"'");
39     System.out.println(sql);
40     try {
41         comandos.executeUpdate(sql);
42     } catch (SQLException ex) {
43         System.out.println("Error en la sintaxis SQL");
44     }
45 }
```

**Método dameLista()**

```
46 public ArrayList<AlumnoPOO> dameLista() {
47     ArrayList<AlumnoPOO> aux= new ArrayList();
48     String sql= "select nombre, apellido, dni from alumnopoo";
49     try {
50         ResultSet tabla=comandos.executeQuery(sql);
51         while (tabla.next()){
52             String nombre= tabla.getString("nombre");
53             String apellido= tabla.getString("apellido");
54             int dni= tabla.getInt("dni");
55             AlumnoPOO unAl= new AlumnoPOO(new Nombre(nombre,apellido),dni,"");
56             aux.add(unAl);
57         }
58     } catch (SQLException ex) {
59         System.out.println("Error "+ ex.toString());
60     }
61     return aux;
62 }
```

**Método dameTabla()**

```
64 public ArrayList<ArrayList<String>> dameTabla() {
65     String sql= "select nombre, apellido, dni from alumnopoo";
66     ArrayList<ArrayList<String>> aux=new ArrayList<ArrayList<String>> ();
67     ResultSet tabla;
68     try {
69         tabla = comandos.executeQuery(sql);
70         while (tabla.next()){
71             String nombre= tabla.getString("nombre");
72             String apellido= tabla.getString("apellido");
73             String dni= tabla.getString("dni");
74             ArrayList<String> fila=new ArrayList<String>();
75             fila.add(nombre);
76             fila.add(apellido);
77             fila.add(dni);
78             aux.add(fila);
79         }
80     } catch (SQLException ex) {
81         System.out.println("ERROR: " + ex.toString());
82     }
83     return aux;
84 }
```

### Método dameDNI()

```
90 public String dameDNI(String apellido) {
91     String dni="";
92     String sql="select dni from alumnopoo where apellido='"+apellido+"'";
93     try {
94         ResultSet registro=comandos.executeQuery(sql);
95         registro.next();
96         dni="" + registro.getInt("dni");
97     } catch (SQLException ex) {
98         Logger.getLogger(AccesoDatos.class.getName()).log(Level.SEVERE, null, ex);
99     }
100     return dni;
101 }
102 }
```

### Método insertarNota()

```
100 public void insertarNota(Nota n) {
101     String sql="insert into nota(dni, trimestre, valor, motivo) values("+n.getDni()+
102         ", "+n.getTrim()+", "+ n.getValor()+ ", '"+ n.getMotivo()+"'";
103     System.out.println(sql);
104     try {
105         comandos.executeUpdate(sql);
106     } catch (SQLException ex) {
107         System.out.println("ERROR: "+ ex.toString());
108     }
109 }
110 }
```