

Completa o corrige las siguientes clases para interactuar con una tabla Alumnos

```
public class Alumno {
//Constante para fijar tamaño máximo en el nombre de alumnos TAM_NOMBRE = 30;
//atributos de alumno id de tipo numérico, nombre, fecha Nacimiento,
notaMedia y curso que es String.

    public Alumno(int id) { }
    public Alumno(id, nombre, fNacimiento, notaMedia, curso) { }

    //Completa los getters y setters que falten

    public Date getfNacimiento() {
        return fNacimiento;
    }

//Genera un método para pintar alumnos con este formato curso-nombre(nota media)

public class AlumnoDao {
    private Connection conexion;
    private final String USUARIO="...";
    private final String PASSWORD="...";
    private final String MAQUINA = "localhost";
    private final String BD = "...";
    public AlumnoDao() {
        conexion = conectar();
    }
    List<Alumno> alumnosXCurso(String curso) { }

    private Connection conectar() { }

    public void create(Alumno alumno) { }

    public Alumno read(int id) {

        String sql = "SELECT * FROM Alumnos WHERE num = ?";
        try {
            PreparedStatement sentencia = conexion.prepareStatement(sql);
            sentencia.setInt(1, id);
            ResultSet rs = sentencia.executeQuery();
        }

    public ... update(Alumno alumno) {
        String sql = "UPDATE Alumnos "
            + "SET nombre=?, fNac=?, media=?, curso=? "
            + "WHERE num = ?";
        sentencia.setInt(5, alumno.getId());
    }
    public void delete(int id) { }
```

```
public static void main(String[] args) {  
    //Ejemplos de uso de las clases y métodos  
  
    AlumnoDao dao = new AlumnoDao();  
  
    List<Alumno> segundoA = dao.alumnosXCurso("2A");  
  
    for(Alumno a: segundoA) {  
        System.out.println(a);  
    }  
  
    Alumno a1 = new Alumno(...);  
    dao.create(a1);  
}
```