

```

1  package org.adrianl.jaxb.libreria;
2  import org.adrianl.jaxb.model.Bookstore;
3
4  import javax.xml.bind.JAXBContext;
5  import javax.xml.bind.JAXBException;
6  import javax.xml.bind.Marshaller;
7  import javax.xml.bind.Unmarshaller;
8  import java.io.File;
9  import java.io.FileNotFoundException;
10 import java.io.FileReader;
11 import java.nio.file.Path;
12 import java.util.ArrayList;
13 public class Main {
14
15     public static void marshal() throws JAXBException {    //Pasar del Objetos a XML
16         //Se crea la lista de libros
17         ArrayList<Libro> libroLista = new ArrayList<>();
18         // Creamos dos libros y los añadimos
19         Libro libro1 = new Libro("Alicia Ramos", "Entornos de Desarrollo", "Garceta"
20 , "978-84-1545-297-3" );
21         libroLista.add(libro1);
22         Libro libro2 = new Libro("Maria Jesús Ramos", "Acceso a Datos", "Garceta",
23 "978-84-1545-228-7" );
24         libroLista.add(libro2);
25         // Se crea La libreria y se le asigna la lista, de libros
26         Libreria milibreria = new Libreria();
27         milibreria.setName("Prueba de libreria JAXB");
28         milibreria.setLugar("Talavera, como no");
29         milibreria.setListaLibro(libroLista);
30         //-----CONVERTIR DE OBJETO A
XML
31         // Creamos el contexto indicando la clase raíz
32         JAXBContext context = JAXBContext.newInstance(Libreria.class);
33         //Creamos el Marshaller, convierte el java bean en una cadena XML
34         Marshaller marshall = context.createMarshaller();
35         //Formateamos el xml para que quede bien
36         marshall.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE
37 );
38         //-----MOSTRAR CONTENIDO
POR TERMINAL
39         // Lo visualizamos con system out
40         marshall.marshal(milibreria, System.out);
41         //-----ESCRITURA DEL
FICHERO
42         // Escribimos en el archivo
43         Path p = Path.of(System.getProperty("user.dir") + File.separator + "data" +
44 File.separator + "libros2.xml");
45         marshall.marshal(milibreria, new File(String.valueOf(p)));
46     }
47
48 }

```

```

45    //Si ahora deseamos hacer lo contrario, es decir, leer los datos del documento
XML y convertirlos a objetos Java
46    public static void unmarshal() throws FileNotFoundException, JAXBException
    {    //PASAR DE XML A OBJETOS
47        //Instanciamos el contexto, indicando la clase que será el RootElement, en
nuestro ejemplo Libreria:
48        JAXBContext context = JAXBContext.newInstance(Libreria.class);
49        //Se crea Unmarshaller en el contexto de la clase Libreria
50        Unmarshaller unmars = context.createUnmarshaller();
51        //Utilizamos el método unmarshal, para obtener datos de un Reader
52        Path uri = Path.of(System.getProperty("user.dir") + File.separator + "data" +
File.separator + "libros.xml");
53        Libreria libreria =(Libreria) unmars.unmarshal(new FileReader(String.valueOf(uri
)));
54        //Recuperamos los datos y visualizamos
55        System.out.println("Nombre de libreria: " + libreria.getNombre());
56        System.out.println("Lugar de la libreria: " + libreria.getLugar());
57        System.out.println("Libros de la librería: ");
58        ArrayList<Libro> lista = libreria.getListLibro();
59
60        lista.stream().map(Libro::getNombre).forEach(System.out::println);
61        lista.forEach((l)->{
62            System.out.println("Título: "+l.getNombre());
63            System.out.println("Autor: "+l.getAutor());
64            System.out.println("Editorial: "+l.getEditorial());
65        });
66    }
67 }
68 /*
69     Una vez que tenemos las clases ya definidas, lo siguiente es ver el código Java
para mapear los objetos
70     que definamos de esas clases. Utilizando la anotación @XmlRootElement. Código
para conseguir el XML
71     //Instanciamos el contexto, indicando la clase que será el RootElement, en nuestro
ejemplo es la clase Libreria
72     JAXBContext jaxbContext = JAXBContext.newInstance(Libreria.class);
73     //Creamos un Marshaller, que es la clase capaz de convertir nuestro JavaBean, en
una cadena XML:
74     Marshaller jaxbMarshaller = jaxbContext.createMarshaller();
75     //Indicamos que vamos a querer el XML con un formato amigable (saltos de línea,
sangrado, etc)
76     jaxbMarshaller.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, true);
77     //Hacemos la conversión llamando al método marshal, pasando una instancia del
JavaBean que queramos convertir
78     // a XML y un OutputStream donde queramos que salga el XML, por ejemplo, la
salida estándar,
79     // o también podría ser un fichero o cualquier otro stream:
80     jaxbMarshaller.marshal(unalInstanciaDeUnaClase, System.out);
81     //Si ponemos un fichero
82     jaxbMarshaller.marshal(unalInstanciaDeUnaClase, new File("./mifichero.xml"));
83     */

```