

```

1  package org.adrianl.sax;
2
3  import org.xml.sax.Attributes;
4  import org.xml.sax.SAXException;
5  import org.xml.sax.helpers.DefaultHandler;
6
7  import java.util.ArrayList;
8  import java.util.List;
9
10 public class NoticiaHandler extends DefaultHandler {
11
12     private boolean hasTitulo;
13     private boolean hasCategoria;
14     private boolean hasDescripcion;
15     private boolean hasFecha;
16     private boolean hasEnlace;
17     private boolean hasAuthor;
18     private boolean enEntry = false;
19     //Almacenamiento de noticias
20
21     private List<Noticia> noticiasList = null;
22     private Noticia noticias = null;
23     public List<Noticia> getNoticias() {
24         return noticiasList;
25     }
26
27     @Override
28     public void startElement(String uri, String localName, String qName, Attributes
attributes) throws SAXException {
29         if (qName.equalsIgnoreCase("item")){
30             //Para que solo obtenga los valores del item y no del channel ya que algunos
como title se repiten.
31             enEntry = true;
32         }
33         if(enEntry){
34             //Inicializar la lista
35             if (noticiasList == null)
36                 noticiasList = new ArrayList<>();
37
38             if (qName.equalsIgnoreCase("item")) {
39                 noticias = new Noticia();
40             } else if (qName.equalsIgnoreCase("title")) {
41                 hasTitulo = true;
42             } else if (qName.equalsIgnoreCase("category")) {
43                 hasCategoria = true;
44             } else if (qName.equalsIgnoreCase("description")) {
45                 hasDescripcion = true;
46             } else if (qName.equalsIgnoreCase("pubDate")) {
47                 hasFecha = true;
48             } else if (qName.equalsIgnoreCase("link")) {
49                 hasEnlace = true;

```

```
50         } else if (qName.equalsIgnoreCase("dc:creator")) {
51             hasAuthor = true;
52         }
53     }
54 }
55
56 @Override
57 public void endElement(String uri, String localName, String qName) throws
SAXException {
58     if (qName.equalsIgnoreCase("item")) {
59         //Se añade cada noticia a la lista
60         noticiasList.add(noticias);
61     }
62 }
63
64 @Override
65 public void characters(char ch[], int start, int length) throws SAXException {
66     //Se añaden los valores de cada noticia
67     if (hasTitulo) {
68         noticias.setTitulo(new String(ch, start, length));
69         hasTitulo = false;
70     } else if (hasCategoria) {
71         noticias.setCategoria(new String(ch, start, length));
72         hasCategoria = false;
73     } else if (hasDescripcion) {
74         noticias.setDescripcion(new String(ch, start, length));
75         hasDescripcion = false;
76     } else if (hasFecha) {
77         noticias.setFecha(new String(ch, start, length));
78         hasFecha = false;
79     } else if (hasEnlace) {
80         noticias.setEnlace(new String(ch, start, length));
81         hasEnlace = false;
82     } else if (hasAuthor) {
83         noticias.setAuthor(new String(ch, start, length));
84         hasAuthor = false;
85     }
86 }
87 }
```