

Ejemplo de implementación de DAO con eXist-db

Vamos a ver un ejemplo parcial de cómo implementar la operación READ por número de departamento sobre la colección **departamentos** y el documento **departamentos.xml** que tenéis en la carpeta de recursos.

El proyecto de ejemplo está en GitHub: https://github.com/add-code-2223/-Ejemplo_eXist_Departamento_Dao_Search-Public.git

En dicho ejemplo tenéis la misma estructura de clases DAO que en la UD2 y UD4, salvo que el método **create** ahora no devuelve el id del departamento sino un boolean indicando si ha habido éxito o no.

En el constructor de **DepartamentoExistDao.java** se inicializan los datos de conexión leyendo el fichero **db.properties** a través de **ConnectionManager**.

Se registra el driver y la base de datos tal y como indica la documentación del API XML:DB

```
DepartamentoExistDao.java
2 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to view and
5 package modelo.dao.departamento;
6
7 import java.io.ByteArrayInputStream;
33
34 /**
35 *
36 * @author mfernandez
37 */
38 public class DepartamentoExistDao extends AbstractGenericDao<Departamento> {
39
40     private static final String DEPT_ROW_TAG = "DEP_ROW";
41     private static final String DEPT_NO_TAG = "DEPT_NO";
42     private static final String DNOMBRE_TAG = "DNOMBRE";
43     private static final String LOC_TAG = "LOC";
44     private MyDataSource dataSource;
45
46     public DepartamentoExistDao() {
47         this.dataSource = ConnectionManager.getDataSource();
48     }
49     try {
50         Class cl = Class.forName(dataSource.getDriver());
51         Database database = (Database) cl.newInstance();
52         database.setProperty("create-database", "true");
53         DatabaseManager.registerDatabase(database);
54
55     } catch (ClassNotFoundException e) {
56         // TODO Auto-generated catch block
57         e.printStackTrace();
58     } catch (InstantiationException e) {
59         // TODO Auto-generated catch block
60         e.printStackTrace();
61     } catch (IllegalAccessException e) {
62         // TODO Auto-generated catch block
63         e.printStackTrace();
64     } catch (XMLDBException e) {
65         // TODO Auto-generated catch block
66         e.printStackTrace();
67     }
68 }
69
70
```

```
db.properties
1 url=xml:db:exist://localhost:8080/e
2 #importante la barra final de url
3 user=admin
4 pwd=abc123.
5 driver=org.exist.xmldb.DatabaseImp
6 coleccionDepartamentos=departament
```

Por el momento solo está implementado el método read():

Toma el nombre de la colección del dataSource, así como el usuario y contraseña.

Utiliza el servicio XQueryService del API de XML:DB y ejecuta la expresión XPath que filtra los elementos DEP_ROW por DEPT_NO=x

```

1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <departamentos>
3   <TITULO>DATOS DE LA TABLA DEPART</TITULO>
4   <DEP_ROW>
5     <DEPT_NO>10</DEPT_NO>
6     <DNOMBRE>CONTABILIDAD</DNOMBRE>
7     <LOC>SEVILLA</LOC>
8   </DEP_ROW>
9   <DEP_ROW>
10    <DEPT_NO>20</DEPT_NO>
11    <DNOMBRE>INVESTIGACION</DNOMBRE>
12    <LOC>MADRID</LOC>
13  </DEP_ROW>

```

```

72 @Override
73 public Departamento read(long id) throws InstanceNotFoundException {
74     Departamento departamento = null;
75
76     try (Collection col = DatabaseManager.getCollection(dataSource.getUrl() + dataSource.getColeccionDepartame
77         dataSource.getUser(), dataSource.getPwd())) {
78
79         XQueryService xqs = (XQueryService) col.getService("XQueryService", "1.0");
80         xqs.setProperty("indent", "yes");
81
82         CompiledExpression compiled = xqs.compile("//DEP_ROW[DEPT_NO=" + id + "]");
83         ResourceSet result = xqs.execute(compiled);
84
85         if (result.getSize() == 0)
86             throw new InstanceNotFoundException(id, Departamento.class.getName());
87
88         ResourceIterator i = result.getIterator();
89         Resource res = null;
90         while (i.hasMoreResources()) {
91             try {
92                 res = i.nextResource();
93
94                 System.out.println(res.getContent().toString());
95
96                 departamento = stringNodeToDepartamento(res.getContent().toString());
97
98             } finally {
99                 // dont forget to cleanup resources
100                 try {
101                     ((EXistResource) res).freeResources();
102                 } catch (XMLDBException xe) {
103                     departamento = null;
104                     xe.printStackTrace();
105                 }
106             }
107         }
108
109     } catch (XMLDBException e) {
110         departamento = null;
111         // TODO Auto-generated catch block
112         e.printStackTrace();
113     }
114     return departamento;
115 }
116

```