

REPORT WORD (DOC Mapping XML)

CODICE GROOVY PER LA CREAZIONE DINAMICA di XML e DOC

STRUTTURA GROOVY

1. recupero dei **dati** dal database che costituiscono la “parte variabile” del documento;
 1. Casistica campi **singoli**
 2. Casistica **liste**
2. creazione della Stringa **xml**
3. recupero del **template Doc** (già caricato in una tabella dei dati)
4. creazione di un **nuovo Doc** a partire da template e xml

Esempio:

```
import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.zip.ZipEntry;
import java.util.zip.ZipFile;
import java.util.zip.ZipOutputStream;

import javax.servlet.http.HttpServletResponse;

import org.apache.poi.util.IOUtils;

import com.geowebframework.transfer.objects.webclient.GwBeanDocument;

//xml da produrre
String xml="";

//1) recupero dei dati
String keyColumn = "id_test";
String itemId = (String) item.get(keyColumn);

HashMap<String, Object> valuesMapQuery = new HashMap<String, Object>();
valuesMapQuery.put(keyColumn,itemId);

String xmlValue ="questo è un test";
```

//1.1)RECUPERO DEL VALORE DI UN CAMPO DAL DB

```
String queryCitta = ""select case when descr_city is null then ' ' else  
descr_city end as descr_city from  
msc_contractor inner join msc_tab_istat_city on  
msc_contractor.cod_cat_city=msc_tab_istat_city.cod_cat_city where  
cod_contractor=#{map.id_contractor}"";
```

```
def cittaRes=services.queryService.executeQuery(queryCitta ,valuesMapQuery)  
String citta = null;  
if(cittaRes!=null && cittaRes.size()>0 && cittaRes.get(0)!=null){  
    citta = cittaRes.get(0).get("descr_city");  
}
```

//1.2)CREAZIONE DI UNA LISTA A APRTIRE DAI RECORD MULTIPLI

//stringa di appoggio per una lista di elementi

```
String xml_riferimenti="";
```

```
String queryRiferimenti=""select case when number_order is null then '  
else number_order end as number_order, case when signing_date_order is null  
then ' ' else signing_date_order end as signing_date_order from msc_estimate  
inner join msc_contractor on  
msc_estimate.id_contractor=msc_contractor.id_contractor where  
cod_contractor=#{map.id_contractor} and estimate_status='C' and  
phase_estimate <>'ANN'"";
```

```
def recordRiferimenti =  
services.queryService.executeQuery(queryRiferimenti,valuesMapQuery);  
String numero_ordine = "";  
String data_ordine = "";
```

```
DateFormat df = new SimpleDateFormat("dd/MM/yyyy");
```

```
for(int j=0; j<recordRiferimenti.size(); j++){  
    numero_ordine = recordRiferimenti[j].number_order;  
    data_ordine = df.format(recordRiferimenti[j].signing_date_order);  
  
    xml_riferimenti+="""<ListaRiferimenti>  
        <numero_ordine>${numero_ordine}</numero_ordine>  
        <data_ordine>${data_ordine}</data_ordine>  
    </ListaRiferimenti>""";  
}
```

//2) creazione della Stringa xml;

//

```
xml =""<RelazioneTrimestrale>  
    <citta>${citta}</citta>  
    <Liste>  
        ${xml_riferimenti}
```

```
</Liste>
</RelazioneTrimestrale>""";

//3) recupero del template Doc
////////////////////////////////////
// classe TEMPLATE DOC
String classNameTemplate = "nome_classe_template";
String attributeColumnName = "nome_attributo_document_template";
////////////////////////////////////
GwBeanDocument gwBeanDocument =
services.gwDocumentDataService.getDocumentBeanByItemIdAndAttributeColumnName
(classNameTemplate,itemId,attributeColumnName);
//nome della cartella per i file temporanei
String tempDestDirName = "gwdocxmltemp";
File tempFile = null;
if(gwBeanDocument!=null && gwBeanDocument.getIs()!=null ){
    File tempDir = new File(System.getProperty("java.io.tmpdir"),
tempDestDirName);
    String tempDirPath = tempDir.getPath();
    log.debug("GEOWEB_DEBUG: If doesn't already exist, trying to create the
directory named by this path:"+tempDirPath);
    if (!tempDir.exists()) {
        if (!tempDir.mkdir())
            throw new IOException("Could not create directory: " +
tempDirPath);
        else
            log.debug("GEOWEB_DEBUG: (FileManagerService) - The folder in the
path " + tempDirPath + " was created");
    }
    tempFile = new File(tempDir,gwBeanDocument.getName());
    FileOutputStream output = new FileOutputStream(tempFile);
    IOUtils.copy(gwBeanDocument.getIs() , output);
}else{
    throw new RuntimeException("Template doc non trovato");
}

//4) SERVIZIO PER LA CREAZIONE DEL DOC AGGIORNATO:
ZipFile zipFile = new ZipFile(tempFile);
OutputStream out = httpServletResponse.getOutputStream();
ZipOutputStream zos = new ZipOutputStream(out);

for(Enumeration e = zipFile.entries(); e.hasMoreElements(); ) {
    ZipEntry entryIn = (ZipEntry) e.nextElement();
    if (!entryIn.getName().equalsIgnoreCase("customXml/item1.xml")) {
        zos.putNextEntry(new ZipEntry(entryIn.getName()));
        InputStream is = zipFile.getInputStream(entryIn);
        IOUtils.copy(is,zos);
    }
    else{
        zos.putNextEntry(new ZipEntry("customXml/item1.xml"));
        byte[] source = (xml).getBytes("UTF-8");
```

```
        ByteArrayInputStream bis = new ByteArrayInputStream(source);
        IOUtils.copy(bis,zos);
    }
    zos.closeEntry();
}
zos.close();
```

From:
<https://wiki.geowebframework.com/> - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=gwtipstricks:idxipstricks:doc_mapping_xml&rev=1616004600

Last update: **2021/03/17 19:10**

