

# OpenOffice integráció

Aki már próbálkozott OpenOffice integrációval, annak hasznos lehet az ODFToolKit ismerete, mivel ehhez nem kell telepített OpenOffice a gépre, hanem az ODF szabvány ismeretében - némi XPath és XML tudással - képesek vagyunk olvasni és írni az OpenOffice által használt dokumentumokat. Az <http://odftoolkit.org> oldalon egész szerteágazó eszközkészlettel találkozunk, nézzük meg közelebbről az ODFDOM csomagot. Az ODFToolKit alapja az ODFDOM, amely egyszerűen betölti a megadott dokumentumot egy DOM fába, majd abban kedvünkre bókálászhatunk, csak ismernünk kell az dokumentumok belső formátumát. Példaképpen hozzunk létre OpenOffice segítségével egy új ODT (szöveg) állományt, írjuk bele valamit, majd mentjük el a */tmp/test.odt* helyre.

Kedvenc Java szerkesztőnként használva hozzunk létre egy új projektet, abba vegyük fel az *odfdom.jar* és a *xerces.jar* állományokat, mint külső könyvtárat, majd írjuk meg az alábbi programrészletet:

```
try
{
    OdfDocument odfDoc = OdfDocument.loadDocument(new File("/tmp/test.odt"));
    OdfFileDom odfContent = odfDoc.getContentDom();

    XPath xpath = odfDoc.getXPath();
    OdfParagraphElement para = (OdfParagraphElement) xpath.evaluate("//text:p[1]", odfContent, XPathConstants.
NODE);
    System.out.println(para.getAttributeNode("text:style-name").getNodeValue());
    System.out.println(para.getFirstChild().getNodeValue());

    para.getFirstChild().setNodeValue("A javaforum.hu csapat a itt járt... :)");
    odfDoc.save(new File("/tmp/test.odt"));
} catch (Exception ex)
{
    Logger.getLogger(Main.class.getName()).log(Level.SEVERE, null, ex);
}
```

A program felépítése logikus és egyszerű: megnyitjuk a fájlt, kérünk rá egy XPath értelmezett megkeressük az első bekezdést, kiírjuk annak stílusát és tartalmát, majd új értéket adunk neki és elmentjük. A módosított fájlt megnyithatjuk OpenOffice segítségével és benne látnunk kell a módosított szöveget.