

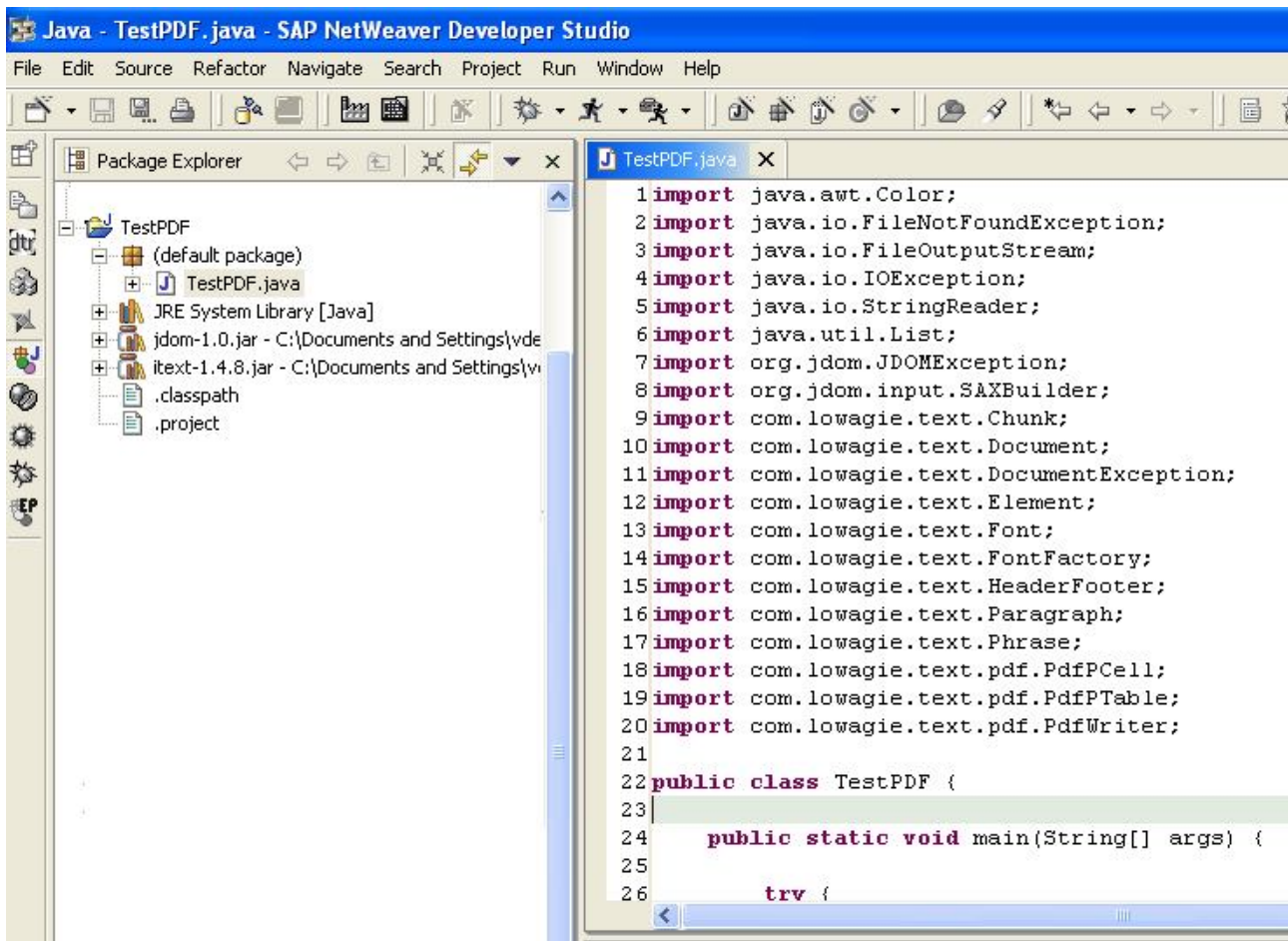
In questo piccolo tutorial mostriamo come integrare in un progetto JAVA la libreria JDOM e la libreria IText.

JDOM è una rappresentazione JAVA di un documento XML; essa fornisce un modo per rappresentare tale documento in modo facile ed efficiente sia in lettura e scrittura, sia in manipolazione. JDOM è un'alternativa a DOM e SAX. E' rilasciata sotto l'Apache-style open source license (la clausola "acknowledgment" è stata rimossa).

Lowagie Itext, rilasciata come free software sotto licenza GNU Affero General Public License version 3, è una libreria che permette la creazione e la manipolazione dinamica di documenti pdf.

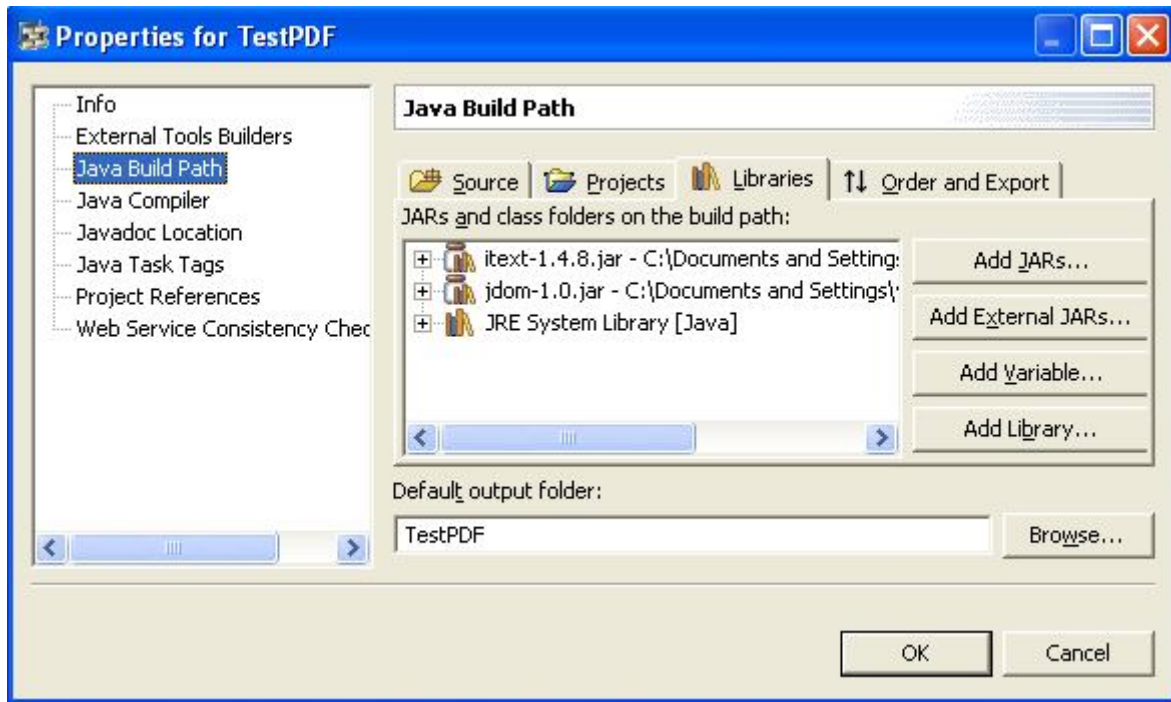
Con l'ausilio di NWDS NetWeaver Developer Studio creiamo un nuovo progetto JAVA (utilizziamo la JAVA Perspective).

Creiamo la classe TestPDF (specifichiamo un pacchetto per il sorgente altrimenti verrà utilizzato il pacchetto default) e specificiamo (tramite checkbox) l'aggiunta del metodo main in modo da poter eseguire la classe in maniera indipendente.



NWDS TestPDF.java

Aggiungiamo (tasto destro sul nome del progetto -> properties ) al JAVA Build Path i jar relativi a JDOM e IText scaricabili da [jdom.org](http://jdom.org) e [itextpdf.com](http://itextpdf.com).



Supponiamo che la nostra fonte dati sia una stringa contenente dati in formato XML.

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE ROOT [<!ELEMENT ROOT ( C* )>
<!ELEMENT C ( P* )>
<!ELEMENT P (#PCDATA)>
<!ATTLIST C CAMPO1 CDATA #REQUIRED CAMPO2 CDATA #REQUIRED CAMPO3 CDATA
#REQUIRED>
<!ATTLIST P CAMPO1 CDATA #REQUIRED CAMPO2 CDATA #REQUIRED CAMPO3 CDATA
#REQUIRED]>

<ROOT>"
  <C CAMPO1='00000001' CAMPO2 ='Gruppo Utenti A' CAMPO3 ='ROMA LAZIO'>
    <P CAMPO1='00000012' CAMPO2 ='FALCO LUDOVICO' CAMPO3 ='ROMA RM' />
    <P CAMPO1='00000014' CAMPO2 ='PICCIONE LUIGI' CAMPO3 ='LATINA LT' />
  </C>
  <C CAMPO1='00000399' CAMPO2 ='Gruppo Utenti B' CAMPO3 ='TORINO PIEMONTE'>
    <P CAMPO1='00000405' CAMPO2 ='AGNELLI SAMUELE' CAMPO3 ='TORINO TO' />
    <P CAMPO1='00000407' CAMPO2 ='ROSSI GIUSEPPE' CAMPO3 ='MILANO MARITTIMA RI' />
  </C>
</ROOT>
```

L'XML in figura è molto semplice. La grammatica definisce un elemento "ROOT" che contiene elementi "C"; gli elementi "C" a loro volta contengono elementi "P". Le "ATTLIST" definiscono la lista degli attributi per ciascun tipo di nodo.

Il codice all'interno del metodo main definisce prima di tutto un documento pdf, in particolare, se ne può definire la grandezza, i margini e altri attributi quali ad esempio, header e footer. Esso lavora come un container per i capitoli, le sezioni, i paragrafi ed il contenuto in genere, di un documento pdf. Il successivo

utilizzo di PdfWriter ci permette di utilizzare l'OutputStream specificato come output del documento (response.getOutputStream() in caso di servlet). Infine, il documento viene aperto attraverso il metodo open().

Utilizziamo una stringa come input per l'XML, allo stesso modo si potrebbe utilizzare un file.

Otteniamo un'istanza SAXBuilder ed un documento org.jdom.Document con l'ausilio di StringReader().

Attraverso l'utilizzo di due cicli for otteniamo le informazioni relative agli attributi dei nodi di tipo "C" (ciclo esterno, metodo *getAttributeValue*) e le informazioni relative agli attributi di tipo "P" (ciclo interno, metodo *getAttributeValue*). Si utilizza il metodo *getChildren()* per ottenere una lista dei figli di un nodo.

Istanziamo oggetti "Table" e "cell" definiamone attributi quali background color, span o margini, ed oggetti "Paragraph", "Phrase" e "Chunk" per le parti testuali (i chunk sono le parti più piccole di testo che possono essere aggiunte ad un documento).

```
import java.awt.Color;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.StringReader;
import java.util.List;
import org.jdom.JDOMException;
import org.jdom.input.SAXBuilder;
import com.lowagie.text.Chunk;
import com.lowagie.text.Document;
import com.lowagie.text.DocumentException;
import com.lowagie.text.Element;
import com.lowagie.text.Font;
import com.lowagie.text.FontFactory;
import com.lowagie.text.HeaderFooter;
import com.lowagie.text.Paragraph;
import com.lowagie.text.Phrase;
import com.lowagie.text.pdf.PdfPCell;
import com.lowagie.text.pdf.PdfPTable;
import com.lowagie.text.pdf.PdfWriter;

public class TestPDF {

    public static void main(String[] args) {
        try {
            Document document = new Document();
            PdfWriter.getInstance(document, new
FileOutputStream("C:/JDOM_IText.pdf"));
            document.addTitle("JDOM e IText");
            HeaderFooter header = new HeaderFooter(new Phrase("JDOM e
IText"), false);
            HeaderFooter footer = new HeaderFooter(new Phrase("JDOM e
IText"), false);
            footer.setAlignment(Element.ALIGN_CENTER);
            document.setHeader(header);
            document.setFooter(footer);
            document.open();
            String xml2 =
                "<?xml version='1.0' encoding='utf-8'?>"
                + "<!DOCTYPE ROOT [<!ELEMENT ROOT ( C* )>"
                + "<!ELEMENT C ( P* )>"
```

```

        + "<!ELEMENT P (#PCDATA)>"
        + "<!ATTLIST C CAMPO1 CDATA #REQUIRED CAMPO2 CDATA
#REQUIRED CAMPO3 CDATA #REQUIRED>"
        + "<!ATTLIST P CAMPO1 CDATA #REQUIRED CAMPO2 CDATA
#REQUIRED CAMPO3 CDATA #REQUIRED]>"
        + "<ROOT>"
        + "<C CAMPO1='00000001' CAMPO2 ='Gruppo Utenti A'
CAMPO3 ='ROMA LAZIO'>"
        + "<P CAMPO1='00000012' CAMPO2 ='FALCO LUDOVICO'
CAMPO3 ='ROMA RM'></P>"
        + "<P CAMPO1='00000014' CAMPO2 ='PICCIONE LUIGI'
CAMPO3 ='LATINA LT'></P>"
        + "</C>"
        + "<C CAMPO1='00000399' CAMPO2 ='Gruppo Utenti B'
CAMPO3 ='TORINO PIEMONTE'>"
        + "<P CAMPO1='00000405' CAMPO2 ='AGNELLI SAMUELE'
CAMPO3 ='TORINO TO'></P>"
        + "<P CAMPO1='00000407' CAMPO2 ='ROSSI GIUSEPPE'
CAMPO3 ='MILANO MARITTIMA RI'></P>"
        + "</C>"
        + "</ROOT>";

    SAXBuilder builder = new SAXBuilder(true);
    org.jdom.Document doc = builder.build(new
StringReader(xml2));

    org.jdom.Element root = doc.getRootElement();

    List figliR = root.getChildren();
    for (int i = 0; i < figliR.size(); i++) {
        org.jdom.Element figlioR = (org.jdom.Element)
figliR.get(i);

        PdfPTable table = new PdfPTable(2);
        table.setWidthPercentage(50);
        table.setHorizontalAlignment(Element.ALIGN_LEFT);
        PdfPCell cell = new PdfPCell(new Paragraph(new
Phrase(new Chunk("CAMPO1 " + figlioR.getAttributeValue("CAMPO1"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(255, 0,
0))))));

        cell.setBackgroundColor(new Color(173,255,47));
        cell.setColspan(2);
        table.addCell(cell);
        cell = new PdfPCell(new Paragraph(new Phrase(new
Chunk("CAMPO2 " + figlioR.getAttributeValue("CAMPO2"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(0, 0,
0))))));

        cell.setBackgroundColor(new Color(143,188,139));
        table.addCell(cell);
        cell = new PdfPCell(new Paragraph(new Phrase(new
Chunk("CAMPO3 " + figlioR.getAttributeValue("CAMPO3"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(0, 0,
0))))));

        cell.setBackgroundColor(new Color(143,188,139));
        table.addCell(cell);
        List figliC = figlioR.getChildren();
        for (int j = 0; j < figliC.size(); j++) {
            org.jdom.Element figlioC = (org.jdom.Element)
figliC.get(j);

            cell = new PdfPCell(new Paragraph(new Phrase(new
Chunk("CAMPO1 " + figlioC.getAttributeValue("CAMPO1"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(255, 0,
0))))));

```

```
        cell.setBackgroundColor(new Color(135,206,250));
        cell.setColspan(2);
        table.addCell(cell);
        cell = new PdfPCell(new Paragraph(new Phrase(new
Chunk("CAMPO2 " + figlioC.getAttributeValue("CAMPO2"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(0, 0,
0))))));
        cell.setBackgroundColor(new Color(173,216,230));
        table.addCell(cell);
        cell = new PdfPCell(new Paragraph(new Phrase(new
Chunk("CAMPO3 " + figlioC.getAttributeValue("CAMPO3"),
FontFactory.getFont(FontFactory.HELVETICA, 7, Font.BOLD, new Color(000, 0,
0))))));
        cell.setBackgroundColor(new Color(173,216,230));
        table.addCell(cell);
    }

    table.setSpacingBefore(5f);
    table.setSpacingAfter(10f);
    document.add(table);
}

    document.close();
} catch (DocumentException e1) { System.out.println("errore " +
e1.getMessage());
} catch (FileNotFoundException e2) { System.out.println("errore " +
e2.getMessage());
} catch (IOException e3) { System.out.println("errore " +
e3.getMessage());
} catch (JDOMException e4) { System.out.println("errore " +
e4.getMessage());
}
}
}
```

TestPDF.java

Il risultato finale:

---

## JDOM e IText

---

<b>Codice Gruppo 00000001</b>	
<b>Sigla Gruppo Gruppo Utenti A</b>	<b>Luogo Gruppo ROMA LAZIO</b>
<b>Codice persona 00000012</b>	
<b>Nome FALCO LUDOVICO</b>	<b>Provenienza ROMA RM</b>
<b>Codice persona 00000014</b>	
<b>Nome PICCIONE LUIGI</b>	<b>Provenienza LATINA LT</b>

<b>Codice Gruppo 00000399</b>	
<b>Sigla Gruppo Gruppo Utenti B</b>	<b>Luogo Gruppo TORINO PIEMONTE</b>
<b>Codice persona 00000405</b>	
<b>Nome AGNELLI SAMUELE</b>	<b>Provenienza TORINO TO</b>
<b>Codice persona 00000407</b>	
<b>Nome ROSSI GIUSEPPE</b>	<b>Provenienza MILANO MARITTIMA RI</b>

risultato finale *JDOM Itext.pdf*

Riferimenti e approfondimenti:

<http://itextpdf.com/>

<http://www.jdom.org/>