

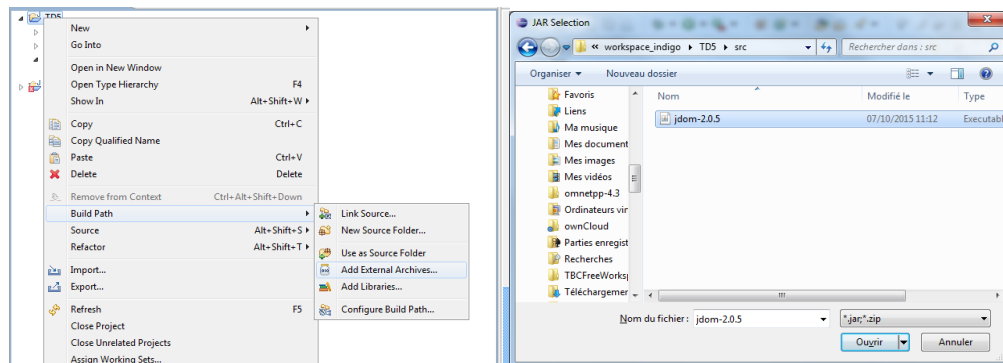
## M3103 - ALGORITHMIQUE AVANCEE JAVA - TP N°5

L'objectif de ce TP est de manipulation des fichiers XML.

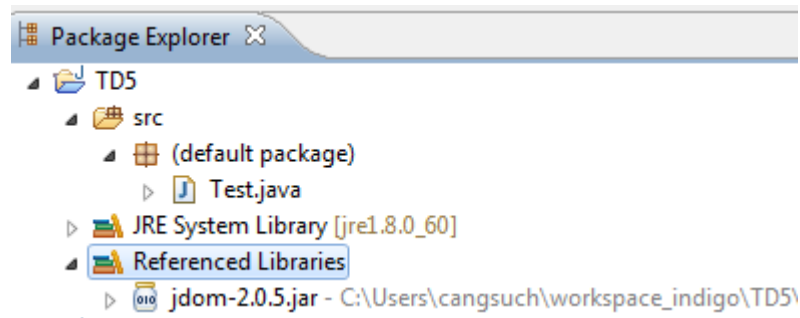
### 1 Ajouter une bibliothèque sur Eclipse

JDOM (Java Document Object Model) est une bibliothèque open source pour manipulation des fichiers XML. Pour cela vous récupérerez sur <http://www.jdom.org/dist/binary/> le fichier jdom-2.0.6.zip. Pour ajouter cette bibliothèque, vous extrayez le fichier zip dans le dossier de votre projet et ajoutez « jdom-2.0.6.jar » sur le projet suivant les instructions ci-dessous:

- Cliquez sur le bouton droit sur votre projet dans le package explorer et choisissez « Build Path -> Add External Archives »
- Précisez la localisation de « jdom-2.0.6.jar » et choisissez le dans la fenêtre « JAR Selection »



- Une fois ajoutée, la bibliothèque va apparaître dans votre projet comme « Referenced Libraries ».



### 2 Comprendre un code de création d'un fichier XML

Reproduire le code vu sur la feuille de TD de la classe « MonPremierFichierXML » pour générer le fichier XML

```
<personnes>
  <etudiant classe="P2">
    <nom>CynO</nom>
  </etudiant>
</personnes>
```

Vérifier que l'affichage est correct et que le fichier Exercice1.XML est correctement créé

```
import java.io.*;
import org.jdom2.*;
import org.jdom2.output.*;

public class MonPremierFichierXML {
    //Nous allons commencer notre arborescence en créant la racine XML qui sera ici "personnes".
    static Element racine = new Element("personnes");

    //On crée un nouveau Document JDOM basé sur la racine que l'on vient de créer
    static org.jdom2.Document document = new Document(racine);

    static void affiche() {
        try {
            //On utilise ici un affichage classique avec getPrettyFormat()
            XMLOutputter sortie = new XMLOutputter(Format.getPrettyFormat());
            sortie.output(document, System.out);
        }
        catch (java.io.IOException e){}
    }

    static void enregistre(String fichier) {
        try {
            // On utilise ici un affichage classique avec getPrettyFormat()
            XMLOutputter sortie = new XMLOutputter(Format.getPrettyFormat());

            // Remarquez qu'il suffit simplement de créer une instance de FileOutputStream avec
            // en argument le
            // nom du fichier pour effectuer la sérialisation.
            sortie.output(document, new FileOutputStream(fichier));
        }
        catch (java.io.IOException e){}
    }

    public static void main(String[] args) {

        //On crée un nouvel Element etudiant et on l'ajoute en tant qu'Element de racine
        Element etudiant = new Element("etudiant");
        racine.addContent(etudiant);

        //On crée un nouvel Attribut classe et on l'ajoute à etudiant grâce à la méthode setAttribute
        Attribute classe = new Attribute("classe", "P2");
        etudiant.setAttribute(classe);

        // On crée un nouvel Element nom, on lui assigne du texte et on l'ajoute en tant
        // qu'Element de etudiant
        Element nom = new Element("nom");
        nom.setText("CynO");
        etudiant.addContent(nom);

        // On affiche le document XML
        affiche();

        // On crée le fichier Exercice1.xml dans lequel on enregistre le document XML
        enregistre("Exercice1.xml");
    }
}
```

### 3 Créer un fichier XML

Compléter ce code java pour créer l'arbre XML ci-dessous

```
<personnes>
  <etudiant classe="P2">
    <nom>CynO</nom>
    <prenoms>
      <prenom>Nicolas</prenom>
      <prenom>Laurent</prenom>
    </prenoms>
  </etudiant>
  <etudiant classe="P1">
    <nom>Superwoman</nom>
  </etudiant>
  <etudiant classe="P1">
    <nom>Don Corleone</nom>
  </etudiant>
</personnes>
```