



 POLITECNICO DI MILANO

Dipartimento di  
Elettronica e Informazione

Planning and Managing Software Projects 2012-13

## Homework 3

**Emanuele Della Valle, Lecturer: Daniele Dell'Aglio**

<http://dellaglio.org>

# Outline

1. Create a new Maven project
  2. Import Apache Commons Lang as dependency
  3. Create a class and write a method
  4. Write a test to verify that the method works as expected
  5. Use Maven to run the tests
  6. Put artefact on the DropBox homework 3 folder
- As explained in the course Web site, the deadline is the 10.6.2013

# POM coordinates

- The project should have the following coordinates:
  - groupId: it.polimi.deib.pmsp.homework3
  - artifactId: group#
    - # is the number of your group. E.g., the group 70 has groupId group70
  - version: 0.1
  - packaging: jar

## The class and method

- Create a class Homework3
- The class must implement the method:  
`String createGroup(int id, String[] components)`
- The method returns a String containing an XML chunk where:
  - The root tag is group
  - The root tag has an attribute number; the value is the method parameter id
  - The root element has a sequence of elements component
    - each element of the components array is associated to a component element

## Example - createGroup

```
String[] components = new String[]{
    "Emanuele Della Valle",
    "Dario Cerizza",
    "Daniele Dell'Aglio"};
String xmlGroup = Homework3.createGroup(0, components);
/*
xmlGroup is:
"<group number="0">
    <component>Emanuele Della Valle</component>
    <component>Dario Cerizza</component>
    <component>Daniele Dell'Aglio</component>
</group>"
*/
```

## Problem: escape!

- The method must return a well formed XML
  - [http://en.wikipedia.org/wiki/Well-formed\\_document](http://en.wikipedia.org/wiki/Well-formed_document)
- In particular:
  - None of the special syntax characters such as "<" and "&" appear except when performing their markup-delineation roles.
- The method should escape the string in components
- The method must escape the strings using `StringEscapeUtils` of the Apache Commons Lang library

# Apache Commons Lang

- The library provides a host of helper utilities for the java.lang API, notably String manipulation methods, basic numerical methods, object reflection, concurrency, creation and serialization and System properties.

[<http://commons.apache.org/proper/commons-lang/>]

- The version to be used is the 3.1
  - Hint: Look for it in <http://mvnrepository.com>
  - Hint: The artifactId is commons-lang3

## escapeXML() method

- The method to be used is

```
public static final String escapeXml(String input)
```

- In the class:

```
org.apache.commons.lang3.StringEscapeUtils
```

[<http://commons.apache.org/proper/commons-lang/apidocs/org/apache/commons/lang3/StringEscapeUtils.html>]

- Examples:

- `StringEscapeUtils.escapeXml("hello")` returns `hello`
- `StringEscapeUtils.escapeXml("hel < lo")` returns `hel &lt; lo`
- `StringEscapeUtils.escapeXml(null)` returns `null`



# Test

- Write the following tests using Junit
  - The createGroup method returns a well-formed XML if the components array contains the element:  
“Mistyped<Component”
  - The createGroup method returns a well-formed XML if components is null
  - createGroup returns a well-formed XML if components is empty

# Submit your work on Dropbox

- Put your work in the homework3 folder in Dropbox
- The folder should contain
  - The Homework3 source file(s) (\*.java)
  - The test file(s) (\*.java)
  - The POM (pom.xml)

## Schema (DTD/XSD)

- ```
<!DOCTYPE group [  
  <!ELEMENT group (component*)>  
  <!ELEMENT component (#PCDATA)>  
  <!ATTLIST group  
    number CDATA #REQUIRED>]>
```
- ```
<xs:element name="group">  
  <xs:attribute name="number"  
type="xs:string"/>  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element name="component"  
type="xs:string" />  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```