Advanced Object Oriented Programming, DT4014, VT24
Assignment 3 (due **May 3**)

Wojciech Mostowski
Halmstad University, ITE

Introduction

You are expected to work as a team in the group you were assigned. Please submit your asnwers through the Blackboard submission system. Include the ZIP file with your code and the short PDF report that summarises your work and answers any "textual" questions that were asked.

The deadline for assignment 3 is **May 3**. The exercises marked with **(*)** must be passed in order to pass the assignment.

Exercises

1. (Exercise 7.4 in the course book)
2. (Exercise 7.10 in the course book)
3. (Exercise 7.19 in the course book)
4. **(*)** Using the annotations and reflection API write an XML file exporter for composite objects of any kind. That is, the exporter should rely only on the annotations in the class definitions of the composite objects to produce the exported XML. The XML tags are defined within the annotations, and otherwise the exporter is **totally indifferent** of the type of the exported object, as long as the required annotations are provided. In particular, there should be no explicit types (for example, `Tree`) mentioned in the code.

   For example, for the following annotated class:

```
@Element(name="node")
public class Tree<T> {

    private Tree<T>[] children = null;
    private T value;

    public Tree(T v, Tree<T>[] trees) { children = trees; value = v; }

    public Tree(T v) { value = v; }

    @SubElements(name="subnodes")
```

```
    public Tree<T>[] getChildren() { return children; }

    @ElementField(name="value")
    public T getValue() { return value; }
}
```

the following program (you have to implement the annotation classes and the `Saver` class with method `save(Object o)`):

```
    Tree<String> t =
      new Tree<String>("top",
        new Tree[] {
          new Tree("sub1"),
          new Tree("sub2")
        });
    Saver s = new Saver();
    String r = s.save(t);
    System.out.println(r);
```

should produce this output:

```
<node value="top">
  <subnodes>
    <node value="sub1"/>
    <node value="sub2"/>
  </subnodes>
</node>
```

It is sufficient that you just create a `String` object with the exported XML, there is no need to implement any file writing routines.