XML and XPath

 The purpose of these exercises is to get initial experience with languages for manipulating and querying XML data. The exercises are adapted from the web programming course at Lund University, taught by Thore Husfeldt. Data for the exercises can be found on the course schedule.

Mindless XPath Drill

This exercise asks you to construct a number of XPath expressions to select specific parts of a the XML document containing recipes. The exercise is supposed to force you through various features of XPath, so you might want to have that book handy. More specifically, write XPath to find:

1. All recipe titles.
2. The name of all ingredients measured in cups.
3. The name of all ingredients measured in tablespoons.
4. The name of all 'immediate' ingredients in recipes (but not of ingredients that are used to make other ingredients).
5. The name of all ingredients that are used to make other ingredients.
6. The name of all ingredients for which you need other ingredients.
7. The title of recipes with eggs in them.
8. The title of recipes with no eggs in them.
9. The name of ingredients where you need exactly half a cup.
10. The name of ingredients where you need more than half a cup.
11. Every second ingredient in the first recipe.

<u>Setup at ITU</u>: Go to `http://exist.itu.dk:8080/exist/sandbox/sandbox.xql`
You can run XPath queries on recipes, using the following syntax:

```
declare default element namespace "http://www.brics.dk/ixwt/recipes";
doc("recipes.xml")<Your XPath expression>
```

For example, to find all recipe titles use `doc("recipes.xml")//title`.

<u>Alternative setup (on your own machine)</u>: "linkedcollection.xml", available in the course schedule, is a version of the the large recipe collection of *An Introduction to XML and Web Technologies* by Møller and Schwarzbach, linked to the XSL stylesheet "selectstuff.xsl".
Copy these two files into a place accessible by your browser. You are supposed to edit the second one. Open the XML file in a browser to see the results of running your XPath expressions.

XSL: Recipes by ingredient

This exercise asks you to construct an XSLT stylesheet for an alternative view of a recipe collection. Create a new copy of the linked recipe collection, and make it point to a stylesheet called "list-of-ingredients.xsl". That is the only modification you may make to "linkedrecipes.xml". Your task is to write "list-of-ingredients.xsl".

Your task. The intended output of your XSL transformation presents the recipe collection *by ingredient*. With each ingredient, the list mentions the recipe(s) in which it appears. The list begins like this:

```
<h1>List of ingredients</h1>
<dl>
<dt>Alchermes liquor</dt>
<dd>Zuppa Inglese</dd>
<dt>angel hair pasta</dt>
<dd>Beef Parmesan with Garlic Angel Hair Pasta</dd>
<dt>baking powder</dt>
<dd>Ricotta Pie (dough)</dd>
</dl>
…
```

I have used an HTML "description list" DD for this, with ingredients in DT elements and recipe names in DD elements.

Warm-up

To begin, solve a simple sub-problem: make a list of all the top-level ingredients in the collection, i.e., those ingredients that are children of "recipe" elements. For example, the list should include "filling" but not "ricotta cheese". Solve this problem in two ways (in two separate stylesheets):
1. Using xsl:apply-templates. This is called Declarative style (or, "Pushing, in XSL slang).
2.Using xsl:for-each. This is called Imperative style (or, "Pulling", in XSL slang).

Further steps

In increasing order of complexity, your final solution needs to display the following features. It's a good idea to solve these problems in this order.
1. The list must include all *non-composite* ingredients, e.g., "ricotta cheese" and even "mushroom juices", but not "filling"
2. The list is sorted
3. The list includes with each ingredient the recipe in which it appears. So "Alechermes liquor" mentions "Zuppa Inglese". (See the above example for a suggestion on formatting–but you can make it look however you want).
4. When an ingredient appears in a composite ingredient, this composite ingredient is named with the recipe. For example, the entry for "mushroom juice" looks like this: mushroom juices Cailles en Sarcophages (sauce)