The product list endpoint available at

[http://redsox.uoa.auckland.ac.nz/ds/DairyService.svc/items](http://redsox.uoa.auckland.ac.nz/ds/DairyService.svc/items Links%20to%20an%20external%20site.)

currently gives a list of products in the following form:

<ArrayOfItem>

<Item>

<ItemId>248309244</ItemId>**integer**

<Origin>Sweden</Origin>

**country name, only a restricted set of strings are permissible**

<Price>19.99</Price>**decimal**

<Title>Julost - Falbygdens - 1059 g</Title>**string**

<Type>Cheese</Type> **string**

</Item>

<Item>

<ItemId>248309246</ItemId>

<Origin>France</Origin>

<Price>16.99</Price>

<Title>ISIGNY Fromage frais à LA FRAMBOISE 500G</Title>

<Type>Yogurt; Cheese</Type>

</Item>

<Item>

<ItemId>248309267</ItemId>

<Origin>France</Origin>

<Price>8.49</Price>

<Title>Fromage frais sucré 7.2%MG</Title>

<Type>Yogurt; Cheese</Type>

</Item>

<Item>

<ItemId>248309268</ItemId>

<Origin>France</Origin>

<Price>7.99</Price>

<Title>Bleu d'Auvergne 100g</Title>

<Type>Cheese</Type>

</Item>

<Item>

<ItemId>248309272</ItemId>

<Origin>Iceland</Origin>

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Type>Yogurt</Type>

</Item>

</ArrayOfItem>

Develop an XML schema that captures this current XML. Your schema should be such that the current XML product list should conform to the schema. Please name your schema file UPI-schema1.xsd where UPI is your UPI.

While most of these fields could be any valid string, the ItemId field should be an integer and the Price field should be a decimal value. In addtion, the Origin field is a country name, therefore only a restricted set of strings are permissible. For example, the following XML product lists should not conform to your schema. (Why?)

<ArrayOfItem>

<Item>

<ItemId>DD248309244</ItemId>

<Origin>Sweden</Origin>

<Price>19.99</Price>

<Title>Julost - Falbygdens - 1059 g</Title>

<Type>Cheese</Type>

</Item>

<Item>

<ItemId>248309272</ItemId>

<Origin>Iceland</Origin>

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Type>Yogurt</Type>

</Item>

</ArrayOfItem>

<ArrayOfItem>

<Item>

<ItemId>248309244</ItemId>

<Origin>Isengard</Origin>

<Price>19.99</Price>

<Title>Julost - Falbygdens - 1059 g</Title>

<Type>Cheese</Type>

</Item>

<Item>

<ItemId>248309272</ItemId>

<Origin>Iceland</Origin>

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Type>Yogurt</Type>

</Item>

</ArrayOfItem>

<ArrayOfItem>

<Item>

<ItemId>248309244</ItemId>

<Origin>Sweden</Origin>

<Price>Free</Price>

<Title>Julost - Falbygdens - 1059 g</Title>

<Type>Cheese</Type>

</Item>

<Item>

<ItemId>248309272</ItemId>

<Origin>Iceland</Origin>

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Type>Yogurt</Type>

</Item>

</ArrayOfItem>

Schema V2

Dunedin Dairy now decides to change the schema such that the ItemId and Origin fields are now attributes. The ItemId attribute is required, while the Origin attribute is optional. If the Origin attribute is omitted, the origin is assumed to be *NZ*.

In addition, the Type field is now replaced with a Types field which contain any number of Type sub-fields. This allows to classify a product into more than one category more elegantly. For example, *Fromage frais sucré* will have the types *Cheese* as well as *Yogurt*. In addition, the types now are restricted to what the shop supports: *Cheese*, *Milk*, *Butter*, *Cream*, and *Yogurt*.

See the examples below.

<ArrayOfItem>

<Item id="248309244" origin="Sweden">

<Price>19.99</Price>

<Title>Julost - Falbygdens - 1059 g</Title>

<Types>

<Type>Cheese</Type>

</Types>

</Item>

<Item id="248309246" origin="France">

<Price>16.99</Price>

<Title>ISIGNY Fromage frais à LA FRAMBOISE 500G</Title>

<Types>

<Type>Cheese</Type>

<Type>Yogurt</Type>

</Types>

</Item>

<Item id="248309267" origin="France">

<Price>8.49</Price>

<Title>Fromage frais sucré 7.2%MG</Title>

<Types>

<Type>Cheese</Type>

<Type>Yogurt</Type>

</Types>

</Item>

<Item id="248309268" origin="France">

<Price>7.99</Price>

<Title>Bleu d'Auvergne 100g</Title>

<Types>

<Type>Cheese</Type>

</Types>

</Item>

<Item id="248309272" origin="Iceland">

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Types>

<Type>Yogurt</Type>

</Types>

</Item>

</ArrayOfItem>

<ArrayOfItem>

<Item id="248309251" origin="Spain">

<Price>12.99</Price>

<Title>Querido Tetilla - Oro del Valle - 650 g</Title>

<Types>

<Type>Cheese</Type>

</Types>

</Item>

<Item id="248309252">

<Price>12.50</Price>

<Title>Anchor Milk Powder Standard Blue bag 1kg</Title>

<Types>

<Type>Milk</Type>

</Types>

</Item>

<Item id="248309253">

<Price>12.50</Price>

<Title>Anchor Milk Powder Trim Milk 1kg</Title>

<Types>

<Type>Milk</Type>

</Types>

</Item>

<Item id="248309254" origin="Sweden">

<Price>11.99</Price>

<Title>Glänta - Arla - 500 g</Title>

<Types>

<Type>Cheese</Type>

</Types>

</Item>

<Item id="248309268" origin="France">

<Price>7.99</Price>

<Title>Bleu d'Auvergne 100g</Title>

<Types>

<Type>Cheese</Type>

</Types>

</Item>

<Item id="248309272" origin="Iceland">

<Price>7.99</Price>

<Title>Ísey Skyr Strawberry 170g</Title>

<Types>

<Type>Yogurt</Type>

</Types>

</Item>

</ArrayOfItem>

Develop an XML schema that captures this new requirement. Your schema should be such that the new XML product lists should conform to the schema. Please name your schema file UPI-schema2.xsd where UPI is your UPI.

Here are some product lists that should not conform to the new XML schema. (Why?)

<ArrayOfItem>

<Item id="248309252">

<Title>Anchor Milk Powder Standard Blue bag 1kg</Title>

<Price>12.50</Price> Title和Price位置反了

<Types>

<Type>Milk</Type>

</Types>

</Item>

<Item id="248309253">

<Title>Anchor Milk Powder Trim Milk 1kg</Title>

<Price>12.50</Price>

<Types>

<Type>Milk</Type>

</Types>

</Item>

</ArrayOfItem>

<ArrayOfItem>

<Item id="248309252">

<Price>12.50</Price>

<Title>Anchor Milk Powder Standard Blue bag 1kg</Title>

信息不完全，少了<Types><Type>…</Type></Types>部分

</Item>

<Item id="248309253">

<Price>12.50</Price>

<Title>Anchor Milk Powder Trim Milk 1kg</Title>

<Types>

<Type>Milk</Type>

</Types>

</Item>

</ArrayOfItem>

<ArrayOfItem>

<Item id="248309252">

<Price>12.50</Price>

<Title>Anchor Milk Powder Standard Blue bag 1kg</Title>

<Types/> 此处不能为empty

</Item>

<Item id="248309253">

<Price>12.50</Price>

<Title>Anchor Milk Powder Trim Milk 1kg</Title>

<Types>

<Type>Milk</Type>

</Types>

</Item>

</ArrayOfItem>

All schemas need to be checked using the XML schema validator supplied to you in the course — online validators may not be robust. Those that do not pass the supplied validators will not attract any mark.

Ensure you form your own additional test cases and test both of your schemas thoroughly.

All schemas are to be hand-crafted. Machine-generated schemas will not attract any mark.

### Submission

Please submit to the [Assignment Dropbox.](https://adb.auckland.ac.nz/) the following items.

1. The first schema file (called UPI-Schema1.xsd where UPI is your UPI).
2. The second schema file (called UPI-Schema2.xsd where UPI is your UPI).

### Additional Tasks

See how you could auto-generate classes that correspond to the schemas you formulated. There are tools to do this. For example, for the C# programming language, check out [XSD](https://manpages.debian.org/stretch/mono-devel/xsd.1.en.html); and for the Java programming language, check out [JAXB](https://javaee.github.io/jaxb-v2/doc/user-guide/ch03.html).

Using these classes you generated, write a program that takes a valid XML instance of a product list as its input, and prints out the name and price of each product in the list.