# WORKSHEET 2-VALIDATION WITH SCHEMAS

1. Create the schema that validates the XML document below. Use types for restrictions:

```
<Customer>
  <IDCust>12345</IDCust>
  <NameCust>Alicia</NameCust>
  <Street>calle del Pez </Street>
  <NumStreet>31</NumStreet>
  <PostalCode>28056</PostalCode>
  <City>Madrid</City>
  <Phone>345-567-153</Phone>
  <DateOfBirth>2001-08-14</DateOfBirth>
  <Discount>true</Discount>
</Customer>
```

a. ID: 5 dígits<br>
b. Name customer: 2 to 8 alphabetic chars<br>
c. Street: String of maximum 30 chars<br>
d. NúmStreet: Integer between 0 and 5000<br>
e. PostalCode: 5 dígits<br>
f. City: Madrid, Barcelona or Sevilla<br>
g. Phone: Pattern: ddd-ddd-ddd<br>
h. DateOfBirth: Date format<br>
i. Discount (True/False): Boolean format

2. Create a valid XML document based on this schema.

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="books" type="BooksForm"/>
  <xsd:complexType name="BooksForm">
    <xsd:sequence>
      <xsd:element name="book" type="BookForm" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="BookForm">
    <xsd:sequence>
      <xsd:element name="author"   type="xsd:string"/>
      <xsd:element name="title"    type="xsd:string"/>
      <xsd:element name="genre"    type="xsd:string"/>
      <xsd:element name="price"    type="xsd:float" />
      <xsd:element name="pub_date" type="xsd:date" />
      <xsd:element name="review"   type="xsd:string"/>
    </xsd:sequence>
    <xsd:attribute name="id"   type="xsd:string"/>
  </xsd:complexType>
</xsd:schema>
```

3. Create the schema that validates the XML document below:

```
<Companies>
  <Company>
    <Name>Bean Shoot Trading</Name>
    <Employee>
      <Name>Joe Bloggs</Name>
      <Age>43</Age>
      <Address>
        <HouseNo>50</HouseNo>
        <Street>Vanilla Crescent</Street>
        <ZipCode>123456</ZipCode>
      </Address>
    </Employee>
    <Employee>
      <Name>Jane Doe</Name>
      <Age>35</Age>
      <Address>
        <HouseNo>1245</HouseNo>
        <Street>Mornington Crescent</Street>
        <ZipCode>66475</ZipCode>
      </Address>
    </Employee>
  </Company>
  <Company>
    <Name>Abacus Inc</Name>
    <Employee>
      <Name>Jane Doe</Name>
      <Age>35</Age>
      <Address>
        <HouseNo>1245</HouseNo>
        <Street>Mornington Crescent</Street>
        <ZipCode>66475</ZipCode>
      </Address>
    </Employee>
    <Employee>
      <Name>Average Joe</Name>
      <Age>26</Age>
        <Address>
        <HouseNo>50</HouseNo>
        <Street>New Street</Street>
        <ZipCode>954674</ZipCode>
      </Address>
    </Employee>
  </Company>
<Companies>
```

4. Create a schema for the xml document and then validate it. Use types for restrictions:

```
<books>
  <book isbn="a1">
    <title type="P">Pepe pepe</title>
    <edition>12</edition>
    <author>
      <first>lqwj</first>
      <middle>r</middle>
      <last>sldfjsl</last>
    </author>
    <author>
      <first>lqwj</first>
      <middle>r</middle>
      <last>sldfjsl</last>
    </author>
    <publishedDate>
      <month>2</month>
      <day>1</day>
      <year>1999</year>
    </publishedDate>
    <type>nonfiction</type>
    <price>003.21</price>
  </book>
  <book isbn="b2">
    <title type="H">SKJFH</title>
    <edition>13</edition>
    <author>
      <first>lqwj</first>
      <middle>r</middle>
      <last>sldfjsl</last>
    </author>
    <publishedDate>
      <month>2</month>
      <day>1</day>
      <year>1979</year>
    </publishedDate>
    <type>fiction</type>
    <price>012.45</price>
  </book>
  <book isbn="c3">
    <title type="H">ALIJSCLK</title>
    <edition>14</edition>
    <author>
      <first>lqwj</first>
      <middle>r</middle>
      <last>sldfjsl</last>
    </author>
```

```
      <publishedDate>
        <month>5</month>
        <day>1</day>
        <year>1989</year>
      </publishedDate>
      <type>fiction</type>
      <price>031.45</price>
    </book>

</books>
```

A. The root element books should have one or more book entries (minimum of one).

B. Each book must contain the following elements, in sequence: title, edition, author (one or more), publishedDate, type, and price

C. Each book contains an isbn number as an ID and is required.

D. title can contain any characters (type of string).

E. title has a type attribute that can contain only the values "P" and "H".

F. edition is an optional element, and must contain only numbers that would never use more than one byte of storage.

G. There may be one or more authors (a minimum of one author) for each book. Each author must have the following child elements first, middle, and last. The elements must be in the order first, middle and last, but middle is an optional element. First, middle and last are all string types, but middle should be at most one character long.

H. The publishedDate month element is defined as a type of unsignedByte with a minimum inclusive value of 1 and a maximum value of 12. The publishedDate day element is defined as a type of unsignedByte with a minimum inclusive value of 1 and a maximum value of 31. The publishedDate year element is defined as a type of gYear

I. The type element is a string and must be either a nonfiction or fiction to be valid

J. The price element is defined as a type of decimal with only three characters to the left of the decimal point and two to the right. The amount must be greater than zero