

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ – ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

«HY452: ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑ ΤΩΝ ΥΠΗΡΕΣΙΩΝ»

ΔΙΔΑΣΚΩΝ: ΝΙΚΟΛΑΟΥ ΧΡΗΣΤΟΣ

ΒΟΗΘΟΣ: ΠΑΠΑΔΑΚΗΣ ΜΥΡΩΝ

19/11/2013

Παράδειγμα

Να φτιάξετε ένα valid **bookstore.xml** και να το συσχετίσετε με το **bookstoreV2.xml**. Το αποτέλεσμα να είναι ο HTML πίνακας του οποίου μέρος φαίνονται παρακάτω:

Bookstore Lab 12

AuthorName	AuthorSurname	Title	Price	Year	isExpensive?
Author6Name	Author6Surname	XML Programming	80 eu		Expensive
Author2Name	Author2Surname	Learning XML	69.95 us	2007	Fair
Author7Name	Author7Surname	XML AND Xpath	50 eu	2009	Cheap

Πιο συγκεκριμένα:

- Θα εμφανίζονται μόνο τα βιβλία που ο τίτλος τους περιέχει την λέξη 'XML'.
- Θα πρέπει τα attributes που υποδεικνύουν την εμφάνιση του table να οριστούν με attribute-set. Πιο συγκεκριμένα θα πρέπει: border = 3, cellpadding = 15, cellspacing = 10, bgcolor=snow.
- Το όνομα και το επώνυμο των συγγραφέων θα πρέπει να φαίνονται σε ξεχωριστές στήλες.
- Στην στήλη price θα εμφανίζεται η τιμή, ένα κενό (με χρήση **variable**) και έπειτα η μονάδα μέτρησης.
- Θα πρέπει να εμφανίζεται» στη στήλη isExpensive?
 1. «Expensive» όταν η τιμή είναι μεγαλύτερη ή ίση του 70,
 2. «Fair» όταν η τιμή είναι μεγαλύτερη από 50 μικρότερη από 70
 3. «Cheap» σε κάθε άλλη περίπτωση.Επίσης βάλετε το κατάλληλο χρώμα ανάλογα με την περίπτωση όπως φαίνεται στην παραπάνω εικόνα.
- Τα βιβλία εμφανίζονται ταξινομημένα σε φθίνουσα σειρά ως προς την τιμή τους.

Solution

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <!--
    Myron Papadakis
    Transformation Services Laboratory
  -->
  <xsl:variable name="space"><xsl:text> </xsl:text></xsl:variable>
  <xsl:template match="/">
    <html>
      <body>
        <h2>Bookstore Lab 12</h2>
        <table xsl:use-attribute-sets="set_table">
          <tr>
            <th>AuthorName</th>
            <th>AuthorSurname</th>
            <th>Title</th>
            <th>Price</th>
            <th>Year</th>
            <th>isExpensive?</th>
          </tr>
          <xsl:for-each select="//book[contains(title,'XML')]">
            <xsl:sort select="price" order="descending" data-type="number"/>
            <tr>
              <td><xsl:value-of select="substring-before(author, ',')" /> </td>
              <td><xsl:value-of select="substring-after(author, ',')" /> </td>
              <td><xsl:value-of select="title" /></td>
              <td><xsl:value-of
select="concat(price,$space,price/@unit)"></xsl:value-of></td>
              <td><xsl:value-of select="year"/></td>

              <!-- Price > 70    : Expensive
                 Price 50-69    : Fair
                 Price <= 50    : Cheap-->
              <xsl:choose>
                <xsl:when test="price >= 70">
                  <td bgcolor="red">Expensive</td>
                </xsl:when>

                <xsl:when test="price>50 and price<70">
                  <td bgcolor="yellow"> Fair </td>
                </xsl:when>
              </td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </template>
</xsl:stylesheet>
```

```

        <xsl:otherwise>
            <td bgcolor="green"> Cheap </td>
        </xsl:otherwise>
    </xsl:choose>
</tr>

</xsl:for-each>
</table>
</body>
</html>
</xsl:template>

<xsl:attribute-set name="set_table">
    <xsl:attribute name="bgcolor">snow</xsl:attribute>
    <xsl:attribute name="border">3</xsl:attribute>
    <xsl:attribute name="cellpadding">15</xsl:attribute>
    <xsl:attribute name="cellspacing">10</xsl:attribute>
</xsl:attribute-set>

</xsl:stylesheet>

```