1. Do the workshop in the chapter 5 & 6
2. After finishing the workshop students must complete the following tasks:
   Cascading Style Sheets
   Extensible Stylesheet Language Transformations
   - Using `<template>`
   - Using `<xsl:value-of>` and `<xsl:for-each>`:

3. Cascading Style Sheets:

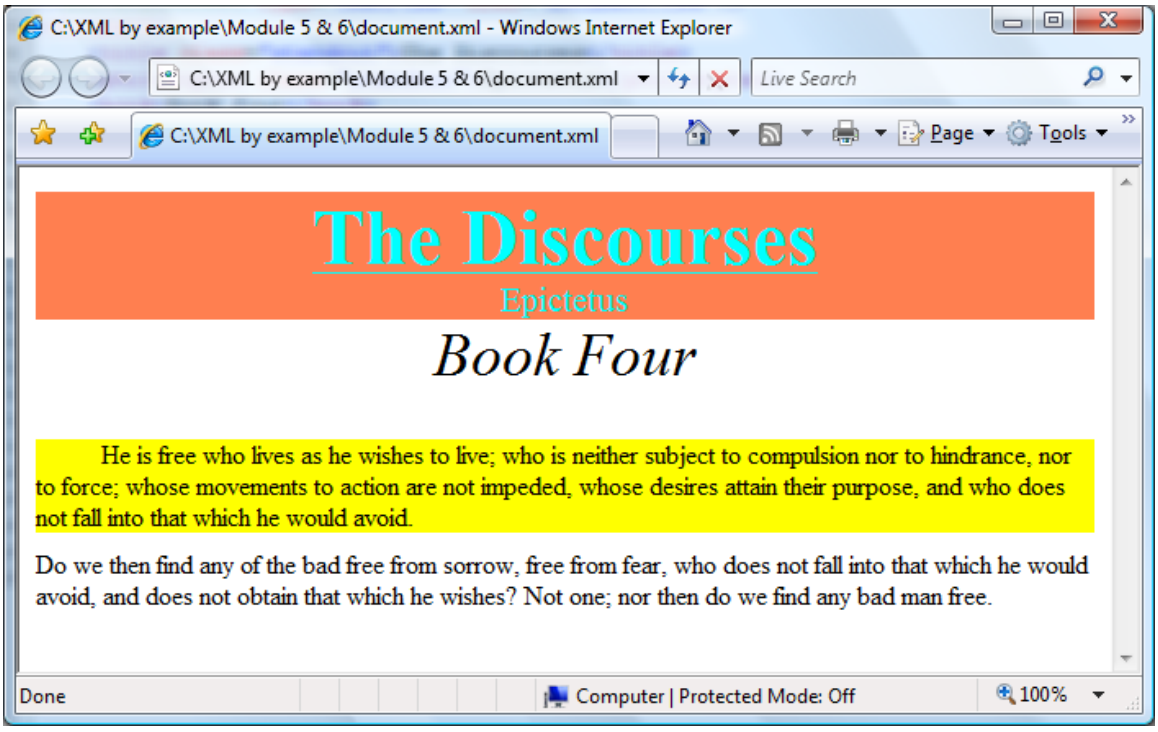
Type and save the following codes as in turn **Document.xml** and **Document.css** files:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/css" href="document.css"?>
<document>
    <title class="standout">The Discourses</title>
    <philosopher class="standout">Epictetus</philosopher>
    <book>Book Four</book>
    <paragraph class="first">
        He is free who lives as he wishes to live; who is neither
        subject to compulsion nor to hindrance, nor to force;
        whose movements to action are not impeded, whose desires
        attain their purpose, and who does not fall into that which
        he would avoid.
    </paragraph>

    <paragraph>
        Do we then find any of the bad free from sorrow, free from
        fear, who does not fall into that which he would avoid, and
        does not obtain that which he wishes? Not one; nor then do
        we find any bad man free.
   </paragraph>
</document>
```

```
title {display: block; font-size: 36pt; font-weight: bold;
    text-align: center; text-decoration: underline}
philosopher {display: block; font-size: 16pt;
    text-align: center}
book {display: block; font-size: 28pt; text-align: center;
    font-style: italic}
paragraph {display: block; margin-top: 10}
.standout {color:cyan; background-color:coral}
paragraph.first {text-indent: 40; margin-top: 30;
background-color:yellow}
```

Open **Document.xml** file in the browser

The output:



4. Extensible Stylesheet Language Transformations

a. Using <template>

Type and save the following codes as in turn **State_data.xml** and **State_data.xsl** files:

```
<?xml version="1.0" encoding ="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="state_data.xsl"?>

<states>

    <state>
        <name>California</name>
        <name>Golden State</name>
        <population units="people">33871648</population><!--2000
census-->
        <capital>Sacramento</capital>
        <bird>Quail</bird>
        <flower>Golden Poppy</flower>
        <area units="square miles">155959</area>
    </state>

    <state>
        <name>Massachusetts</name>
        <name>Bay State</name>
        <population units="people">6349097</population><!--2000 census-->
        <capital>Boston</capital>
        <bird>Chickadee</bird>
        <flower>Mayflower</flower>
        <area units="square miles">7840</area>
```

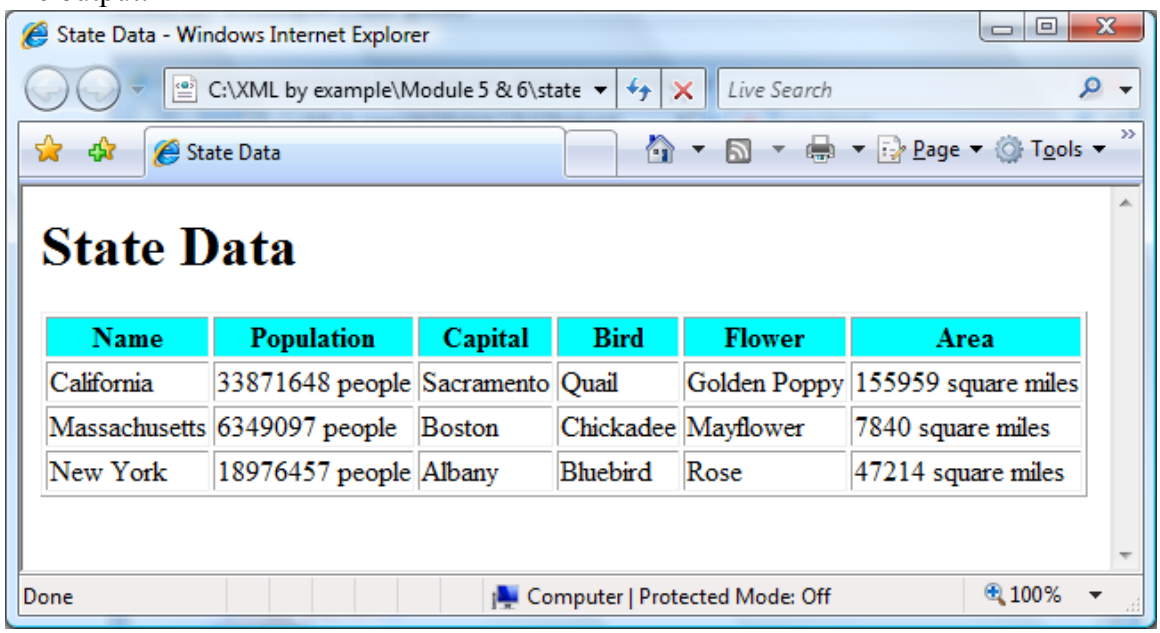
```
    </state>

    <state>
        <name>New York</name>
        <name>Empire State</name>
        <population units="people">18976457</population><!--2000
census-->
        <capital>Albany</capital>
        <bird>Bluebird</bird>
        <flower>Rose</flower>
        <area units="square miles">47214</area>
    </state>

</states>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

    <xsl:template match="/states">
        <HTML>
            <HEAD>
                <TITLE>
                    State Data
                </TITLE>
            </HEAD>
            <BODY>
                <H1>
                    State Data
                </H1>
                <TABLE BORDER="1">
                    <TR BGCOLOR = "CYAN">
                        <TH>Name</TH>
                        <TH>Population</TH>
                        <TH>Capital</TH>
                        <TH>Bird</TH>
                        <TH>Flower</TH>
                        <TH>Area</TH>
                    </TR>
                    <xsl:apply-templates/>
                </TABLE>
            </BODY>
        </HTML>
    </xsl:template>

    <xsl:template match="state">
       <TR>
          <TD><xsl:value-of select="name"/></TD>
          <TD><xsl:apply-templates select="population"/></TD>
          <TD><xsl:apply-templates select="capital"/></TD>
          <TD><xsl:apply-templates select="bird"/></TD>
          <TD><xsl:apply-templates select="flower"/></TD>
          <TD><xsl:apply-templates select="area"/></TD>
       </TR>
   </xsl:template>
```

```
    <xsl:template match="population">
        <xsl:value-of select="."/>
        <xsl:text> </xsl:text>
        <xsl:value-of select="@units"/>
    </xsl:template>

    <xsl:template match="capital">
        <xsl:value-of select="."/>
    </xsl:template>

    <xsl:template match="bird">
        <xsl:value-of select="."/>
    </xsl:template>

    <xsl:template match="flower">
        <xsl:value-of select="."/>
    </xsl:template>

    <xsl:template match="area">
        <xsl:value-of select="."/>
        <xsl:text> </xsl:text>
        <xsl:value-of select="@units"/>
    </xsl:template>

</xsl:stylesheet>
```

Open **State_data.xml** file in the browser

The output:



b. Using `<xsl:value-of>` and `<xsl:for-each>`:

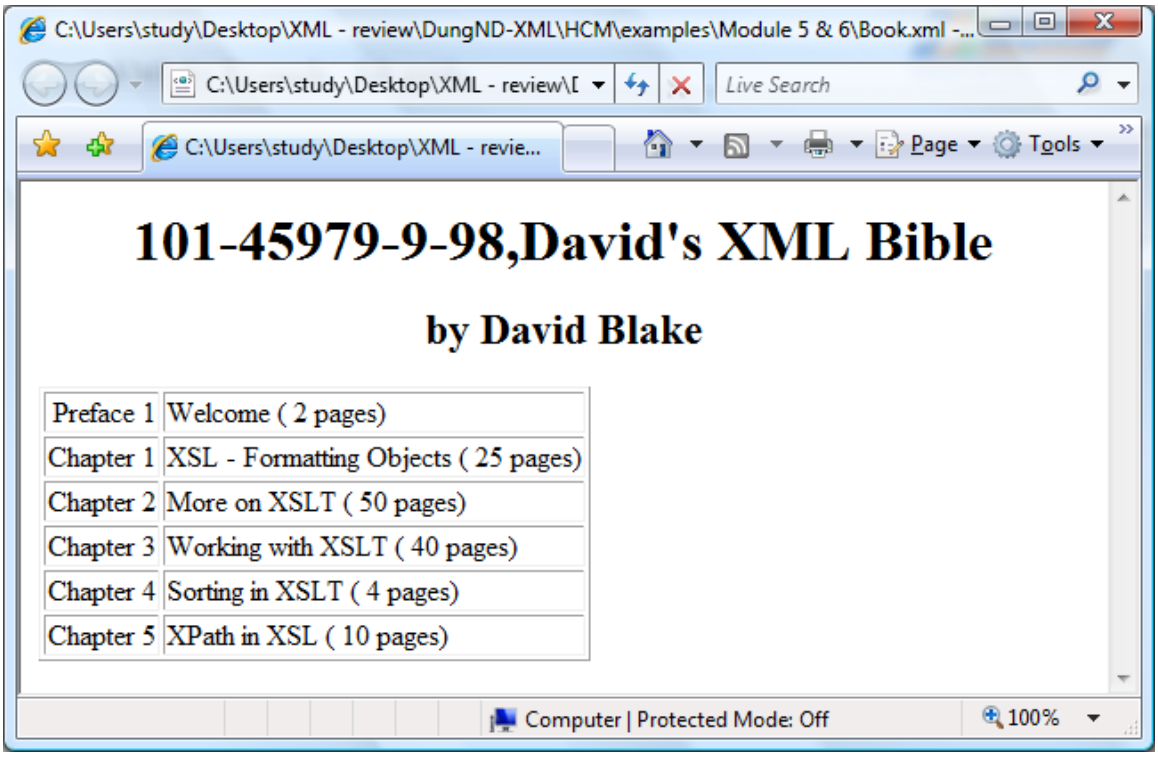
Type and save the following codes as in turn **Book.xml** and **Book.xsl** files:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type = "text/xsl" href = "Book.xsl"?>
<Book>
  <ISBN>101-45979-9-98</ISBN>
 <Title>David's XML Bible</Title>
 <Author>
    <FirstName>David</FirstName>
    <LastName>Blake</LastName>
 </Author>
 <Chapters>
    <Preface>
      <Name>Welcome</Name>
      <Number>1</Number>\
      <Pages>2</Pages>
     </Preface>
        <Chapter>
          <Name>Working with XSLT</Name>
          <Number>3</Number>
          <Pages>40</Pages>
        </Chapter>
        <Chapter>
          <Name>Sorting in XSLT</Name>
        <Number>4</Number>
        <Pages>4</Pages>
      </Chapter>
      <Chapter>
        <Name>More on XSLT</Name>
        <Number>2</Number>
        <Pages>50</Pages>
      </Chapter>
      <Chapter>
        <Name>XPath in XSL</Name>
        <Number>5</Number>
        <Pages>10</Pages>
      </Chapter>
      <Chapter>
        <Name>XSL - Formatting Objects</Name>
        <Number>1</Number>
        <Pages>25</Pages>
      </Chapter>
   </Chapters>
  </Book>
```

```
<?xml version="1.0" encoding="UTF-8" ?>

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:output method="html"/>

  <xsl:template match="/Book">
```

```
    <body>
      <h1 align = "center">
        <xsl:value-of select="ISBN"/>
        <xsl:text>,</xsl:text>

        <xsl:value-of select="Title"/>
      </h1>
      <h2 align = "center">
        <xsl:text>by </xsl:text>
        <xsl:value-of select="Author/FirstName"/>

        <xsl:text> </xsl:text>
        <xsl:value-of select="Author/LastName"/>
      </h2>
      <table border="1">
        <tr>
          <td align="right">
            <xsl:text>Preface </xsl:text>
            <xsl:value-of select="Chapters/Preface/Number"/>

          </td>
          <td>
            <xsl:value-of select="Chapters/Preface/Name"/>
            <xsl:text>  ( </xsl:text>
            <xsl:value-of select="Chapters/Preface/Pages"/>
            <xsl:text>  pages) </xsl:text>
          </td>
        </tr>

        <xsl:for-each select="Chapters/Chapter">
          <xsl:sort select="Number" order="ascending"/>
          <tr>
            <td align="right">
              <xsl:text>Chapter </xsl:text>
              <xsl:value-of select="Number"/>
            </td>
            <td>
              <xsl:value-of select="Name"/>
              <xsl:text>  ( </xsl:text>
              <xsl:value-of select="Pages"/>
              <xsl:text>  pages) </xsl:text>
            </td>
          </tr>
        </xsl:for-each>
      </table>
    </body>
  </xsl:template>
</xsl:stylesheet>
```

The output:



Do It Yourself

3.1. Modify the State_data.xsl file to display in the table sorted according to the data in <Area> tag.

3.2. Transform the following xml document:

```
<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="SalesData_stylesheet.xsl"?>

<CottonDesk>

  <SalesPerson>

    <Product>Monitor</Product>

    <Name>Harry Diana</Name>

    <Age>23</Age>
```

```
    <Sales>4000</Sales>

  </SalesPerson>
  <SalesPerson>

    <Product>Laptop</Product>

    <Name>David Blake</Name>

    <Age>32</Age>

    <Sales>20000</Sales>

  </SalesPerson>

    <SalesPerson>

    <Product>Laptop</Product>

    <Name>Susan Jones</Name>

    <Age>55</Age>

    <Sales>35000</Sales>

  </SalesPerson>

    <SalesPerson>

    <Product>Laptop</Product>

    <Name>Martin Howell</Name>

    <Age>25</Age>

    <Sales>1000</Sales>

  </SalesPerson>

    <SalesPerson>

    <Product>Keyboard</Product>

    <Name>John Dani</Name>

    <Age>45</Age>

    <Sales>35000</Sales>
```

```
  </SalesPerson>

    <SalesPerson>

    <Product>Laptop</Product>

    <Name>Tony Kemp</Name>

    <Age>35</Age>

    <Sales>31000</Sales>

  </SalesPerson>

  </CottonDesk>
```

Into the html document as showed bellow: