

CSIT128 / CSIT828

XSLT

Joseph Tonien

XSL

EXtensible **S**tylesheet **L**anguage (XSL) is a family of languages used to transform and render XML documents:

XSL Transformation (XSLT): a language for transforming XML documents

XML Path Language (XPath): a language for navigating in XML documents

XQuery: a language for querying XML documents

XSLT

EXtensible **S**tylesheet **L**anguage **T**ransformation (XSLT) is an XML language for transforming XML documents

file extension is .xsl

used to transform XML file into other file formats, such as HTML

describes how the XML elements should be displayed.

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

  ... xslt code here ...

</xsl:stylesheet>
```

XSLT

Have a look at the XML file

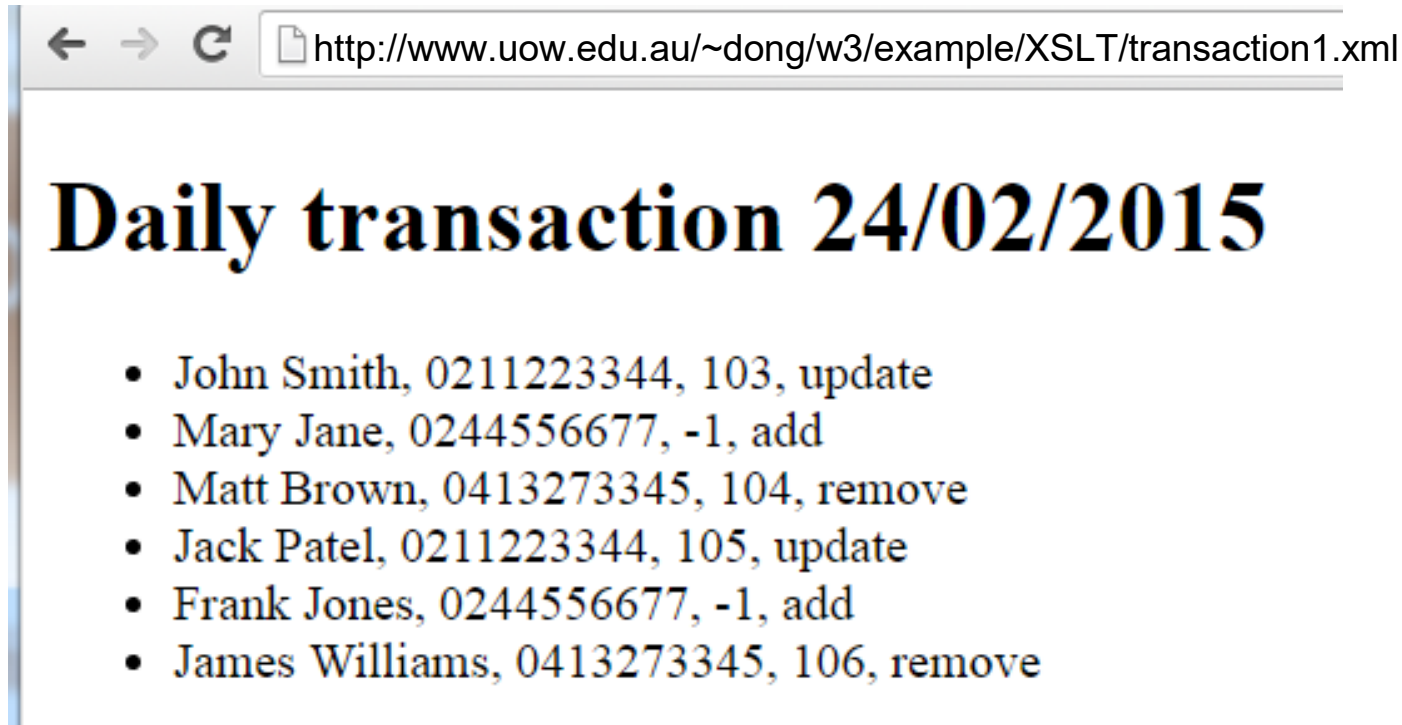
<http://www.uow.edu.au/~dong/w3/example/XSLT/transaction.xml>

```
<?xml version="1.0" encoding="UTF-8" ?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

Example 1: Now have a look at the XML file

<http://www.uow.edu.au/~dong/w3/example/XSLT/transaction1.xml>



XSLT

The content of `transaction1.xml` is exactly the same as `transaction.xml`, except that it uses an xml stylesheet: **`transaction-style1.xsl`**

```
<?xml version="1.0" encoding="UTF-8" ?>
<?xml-stylesheet type="text/xsl" href="transaction-style1.xsl"?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>

  <xsl:template match="/dailyTransaction">
    <html>
      <head>
        <title>XSLT example</title>
      </head>
      <body>
        <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
        <ul>
          <xsl:for-each select="person">
            <li>
              <xsl:value-of select="firstName" />
              <xsl:text> </xsl:text>
              <xsl:value-of select="lastName" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="mobile" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@staffDbId" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@operation" />
            </li>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>

  <xsl:template match="/dailyTransaction">
    <html>
      <head>
        <title>XSLT example</title>
      </head>
      <body>
        <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
        <ul>
          <xsl:for-each select="person">
            <li>
              <xsl:value-of select="firstName" />
              <xsl:text> </xsl:text>
              <xsl:value-of select="lastName" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="mobile" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@staffDbId" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@operation" />
            </li>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="transaction-style1.xsl">
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

Daily transaction 24/02/2015

- John Smith, 0211223344, 103, update
- Mary Jane, 0244556677, -1, add
- Matt Brown, 0413273345, 104, remove
- Jack Patel, 0211223344, 105, update
- Frank Jones, 0244556677, -1, add
- James Williams, 0413273345, 106, remove

XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

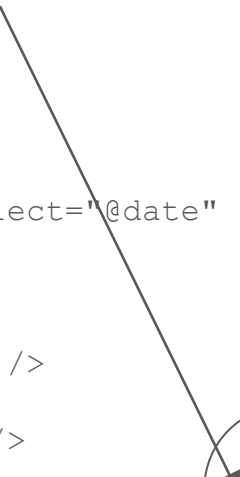
  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>
```

<xsl:template match="/dailyTransaction">

```
  <html>
    <head>
      <title>XSLT example</title>
    </head>
    <body>
      <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
      <ul>
        <xsl:for-each select="person">
          <li>
            <xsl:value-of select="firstName" />
            <xsl:text> </xsl:text>
            <xsl:value-of select="lastName" />
            <xsl:text>, </xsl:text>
            <xsl:value-of select="mobile" />
            <xsl:text>, </xsl:text>
            <xsl:value-of select="@staffDbId" />
            <xsl:text>, </xsl:text>
            <xsl:value-of select="@operation" />

          </li>
        </xsl:for-each>
      </ul>
    </body>
  </html>
</xsl:template>
```

```
</xsl:stylesheet>
```



A diagram consisting of a straight line with an arrowhead pointing from the **<xsl:template match="/dailyTransaction">** line in the XSLT code block to the XML output block.

```
<?xml version="1.0" encoding="UTF-8" ?>
<b>dailyTransaction</b> date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>

  <xsl:template match="/dailyTransaction">
    <html>
      <head>
        <title>XSLT example</title>
      </head>
      <body>
        <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
        <ul>
          <xsl:for-each select="person">
            <li>
              <xsl:value-of select="firstName" />
              <xsl:text> </xsl:text>
              <xsl:value-of select="lastName" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="mobile" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@staffDbId" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@operation" />
            </li>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

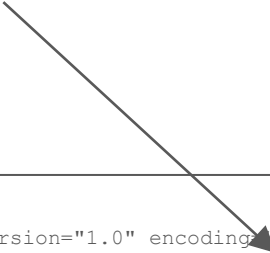
Let's look at the xml stylesheet: **transaction-style1.xsl**

```
<h1>Daily transaction <xsl:value-of select="@date" /> </h1>
```

select="@date"
means the **attribute**

Daily transaction 24/02/2015

- John Smith, 0211223344, 103, update
- Mary Jane, 0244556677, -1, add
- Matt Brown, 0413273345, 104, remove
- Jack Patel, 0211223344, 105, update
- Frank Jones, 0244556677, -1, add
- James Williams, 0413273345, 106, remove



```
<?xml version="1.0" encoding="UTF-8" ?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>

  <xsl:template match="/dailyTransaction">
    <html>
      <head>
        <title>XSLT example</title>
      </head>
      <body>
        <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
        <ul>
          <xsl:for-each select="person">
            <li>
              <xsl:value-of select="firstName" />
              <xsl:text> </xsl:text>
              <xsl:value-of select="lastName" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="mobile" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@staffDbId" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@operation" />
            </li>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```



The diagram illustrates the transformation of the XSLT code into XML output. Two arrows originate from the **<xsl:for-each select="person">** line in the XSLT code. One arrow points to the first **<person>** element in the output XML, and the other points to the second **<person>** element. The output XML is shown in a rounded rectangle on the right side of the image.

```
<?xml version="1.0" encoding="UTF-8" ?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```


XSLT

Let's look at the xml stylesheet: **transaction-style1.xsl**

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

  <xsl:output method="xml" indent="yes" encoding="UTF-8"/>

  <xsl:template match="/dailyTransaction">
    <html>
      <head>
        <title>XSLT example</title>
      </head>
      <body>
        <h1>Daily transaction <xsl:value-of select="@date" /> </h1>
        <ul>
          <xsl:for-each select="person">
            <li>
              <xsl:value-of select="firstName" />
              <xsl:text> </xsl:text>
              <xsl:value-of select="lastName" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="mobile" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@staffDbId" />
              <xsl:text>, </xsl:text>
              <xsl:value-of select="@operation" />
            </li>
          </xsl:for-each>
        </ul>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

```
<ul>
```

```
  <xsl:for-each select="person">
```

```
    <li>
```

```
      <xsl:value-of select="firstName" />
```

```
      <xsl:text> </xsl:text>
```

```
      <xsl:value-of select="lastName" />
```

```
      <xsl:text>, </xsl:text>
```

```
      <xsl:value-of select="mobile" />
```

```
      <xsl:text>, </xsl:text>
```

```
      <xsl:value-of select="@staffDbId" />
```


```
      <xsl:text>, </xsl:text>
```

```
      <xsl:value-of select="@operation" />
```

```
    </li>
```

```
  </xsl:for-each>
```

```
</ul>
```



```
<person staffDbId="103" operation="update">  
  <firstName>John</firstName>  
  <lastName>Smith</lastName>  
  <mobile>0211223344</mobile>  
</person>
```

XSLT

```
<ul>
```

```
<xsl:for-each select="person">
```

```
<li>
```

```
<xsl:value-of select="firstName" />
```

```
<xsl:text> </xsl:text>
```

```
<xsl:value-of select="lastName" />
```

```
<xsl:text>, </xsl:text>
```

```
<xsl:value-of select="mobile" />
```

```
<xsl:text>, </xsl:text>
```

```
<xsl:value-of select="@staffDbId" />
```

```
<xsl:text>, </xsl:text>
```

```
<xsl:value-of select="@operation" />
```

```
</li>
```

```
</xsl:for-each>
```

```
</ul>
```

this is an element

this is an attribute

```
<person staffDbId="103" operation="update">
  <firstName>John</firstName>
  <lastName>Smith</lastName>
  <mobile>0211223344</mobile>
</person>
```

XSLT

```
<ul>
  <xsl:for-each select="person">
    <li>
      <xsl:value-of select="firstName" />
      <xsl:text> </xsl:text>
      <xsl:value-of select="lastName" />
      <xsl:text>, </xsl:text>
      <xsl:value-of select="mobile" />
      <xsl:text>, </xsl:text>
      <xsl:value-of select="@staffDbId" />
      <xsl:text>, </xsl:text>
      <xsl:value-of select="@operation" />
    </li>
  </xsl:for-each>
</ul>
```

Daily transaction 24/02/2015

- John Smith, 0211223344, 103, update
- Mary Jane, 0244556677, -1, add

```
<person staffDbId="103" operation="update">
  <firstName>John</firstName>
  <lastName>Smith</lastName>
  <mobile>0211223344</mobile>
</person>
```

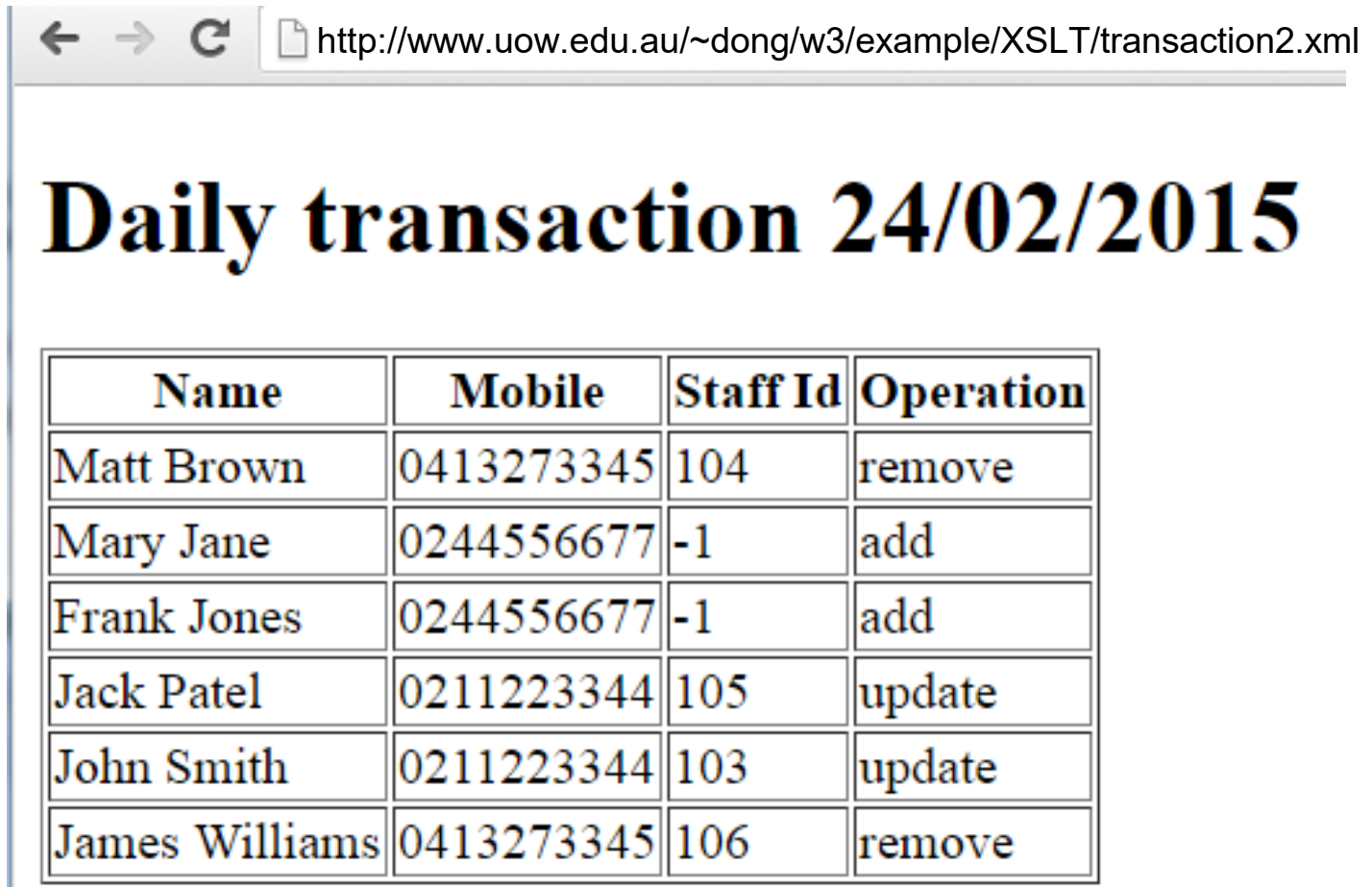

XSLT

Example 2: Now look at `transaction2.xml`, it uses an xml stylesheet:
`transaction-style2.xsl`

```
<?xml version="1.0" encoding="UTF-8" ?>
<?xml-stylesheet type="text/xsl" href="transaction-style2.xsl"?>
<dailyTransaction date="24/02/2015">
  <person staffDbId="103" operation="update">
    <firstName>John</firstName>
    <lastName>Smith</lastName>
    <mobile>0211223344</mobile>
  </person>
  <person staffDbId="-1" operation="add">
    <firstName>Mary</firstName>
    <lastName>Jane</lastName>
    <mobile>0244556677</mobile>
  </person>
  ...
</dailyTransaction>
```

XSLT

View the source of the xml stylesheet: `transaction-style2.xsl`
we can see that it displays a table



A screenshot of a web browser window. The address bar shows the URL `http://www.uow.edu.au/~dong/w3/example/XSLT/transaction2.xml`. The main content area displays a large, bold, black serif title "Daily transaction 24/02/2015". Below the title is a table with four columns: "Name", "Mobile", "Staff Id", and "Operation". The table contains six rows of data.

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

XSLT

View the source of the xml stylesheet: **transaction-style2.xsl**
we can see that it displays a table

```
<xsl:for-each select="person">
  <xsl:sort select="lastName"/>
  <tr>
    ...
  </tr>
</xsl:for-each>
```

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

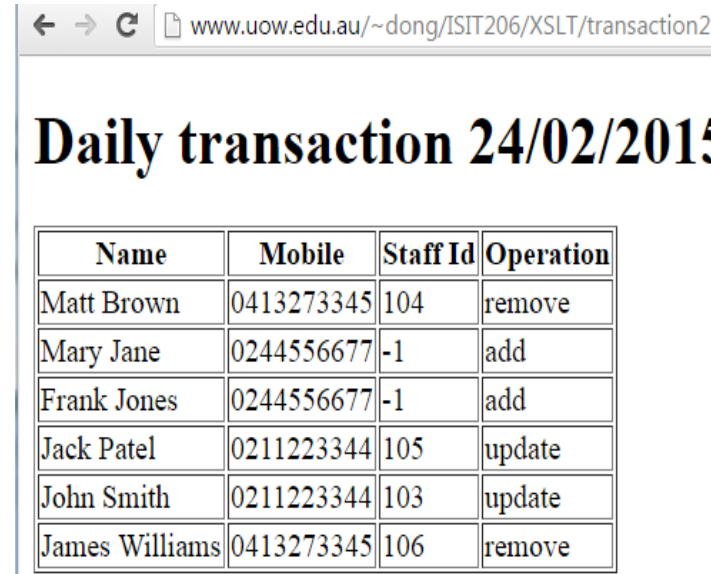
XSLT

Example 3: Compare `transaction-style2.xsl` with `transaction-style3.xsl`

```
<table border="1">
  <tr>
    <th>Name</th>
    <th>Mobile</th>
    <th>Staff Id</th>
    <th>Operation</th>
  </tr>

  <xsl:apply-templates select="person">
    <xsl:sort select="lastName" />
  </xsl:apply-templates>

</table>
```



Daily transaction 24/02/2015			
Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

```

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

<xsl:template match="/">
  <html>
  <body>
  <h2>My CD Collection</h2>
  <xsl:apply-templates/>
  </body>
  </html>
</xsl:template>

<xsl:template match="cd">
  <p>
    <xsl:apply-templates select="title"/>
    <xsl:apply-templates select="artist"/>
  </p>
</xsl:template>

<xsl:template match="title">
  Title: <span style="color:#ff0000">
    <xsl:value-of select="."/></span>
  <br />
</xsl:template>

<xsl:template match="artist">
  Artist: <span style="color:#00ff00">
    <xsl:value-of select="."/></span>
  <br />
</xsl:template>

</xsl:stylesheet>

```

XML Code:

```

<?xml version="1.0" encoding="UTF-8"?>
<catalog>
  <cd>
    <title>Empire Burlesque</title>
    <artist>Bob Dylan</artist>
    <country>USA</country>
    <company>Columbia</company>
    <price>10.90</price>
    <year>1985</year>
  </cd>
  <cd>
    <title>Hide your heart</title>
    <artist>Bonnie Tyler</artist>
    <country>UK</country>

```

Edit the XML or XSLT code above and Click Me »

My CD Collection

Title: **Empire Burlesque**
Artist: **Bob Dylan**

Title: **Hide your heart**
Artist: **Bonnie Tyler**

Title: **Greatest Hits**
Artist: **Dolly Parton**

Title: **Still got the blues**
Artist: **Gary Moore**

Title: **Eros**
Artist: **Eros Ramazzotti**

Title: **One night only**
Artist: **Bee Gees**

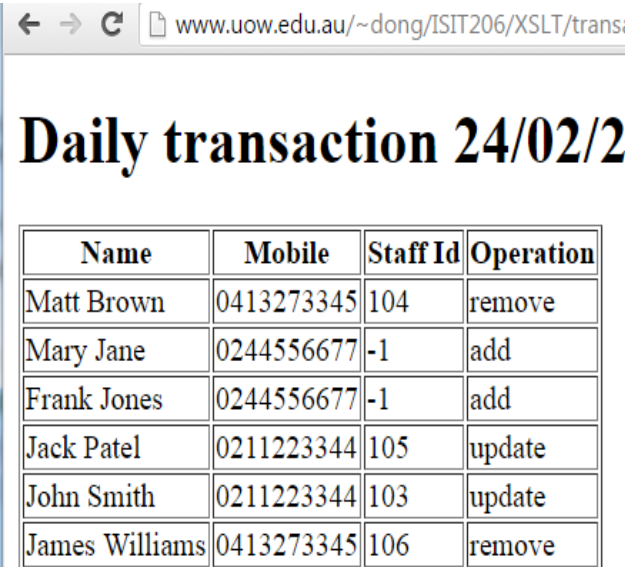
Title: **Sylvias Mother**
Artist: **Dr Hook**

https://www.w3schools.com/xml/xsl_apply_templates.asp

XSLT

Now compare **transaction-style2.xsl** with **transaction-style3.xsl**

```
<xsl:template match="person">
  <tr>
    <td>
      <xsl:value-of select="firstName" />
      <xsl:text> </xsl:text>
      <xsl:value-of select="lastName" />
    </td>
    <td>
      <xsl:value-of select="mobile" />
    </td>
    <td>
      <xsl:value-of select="@staffDbId" />
    </td>
    <td>
      <xsl:value-of select="@operation" />
    </td>
  </tr>
</xsl:template>
```



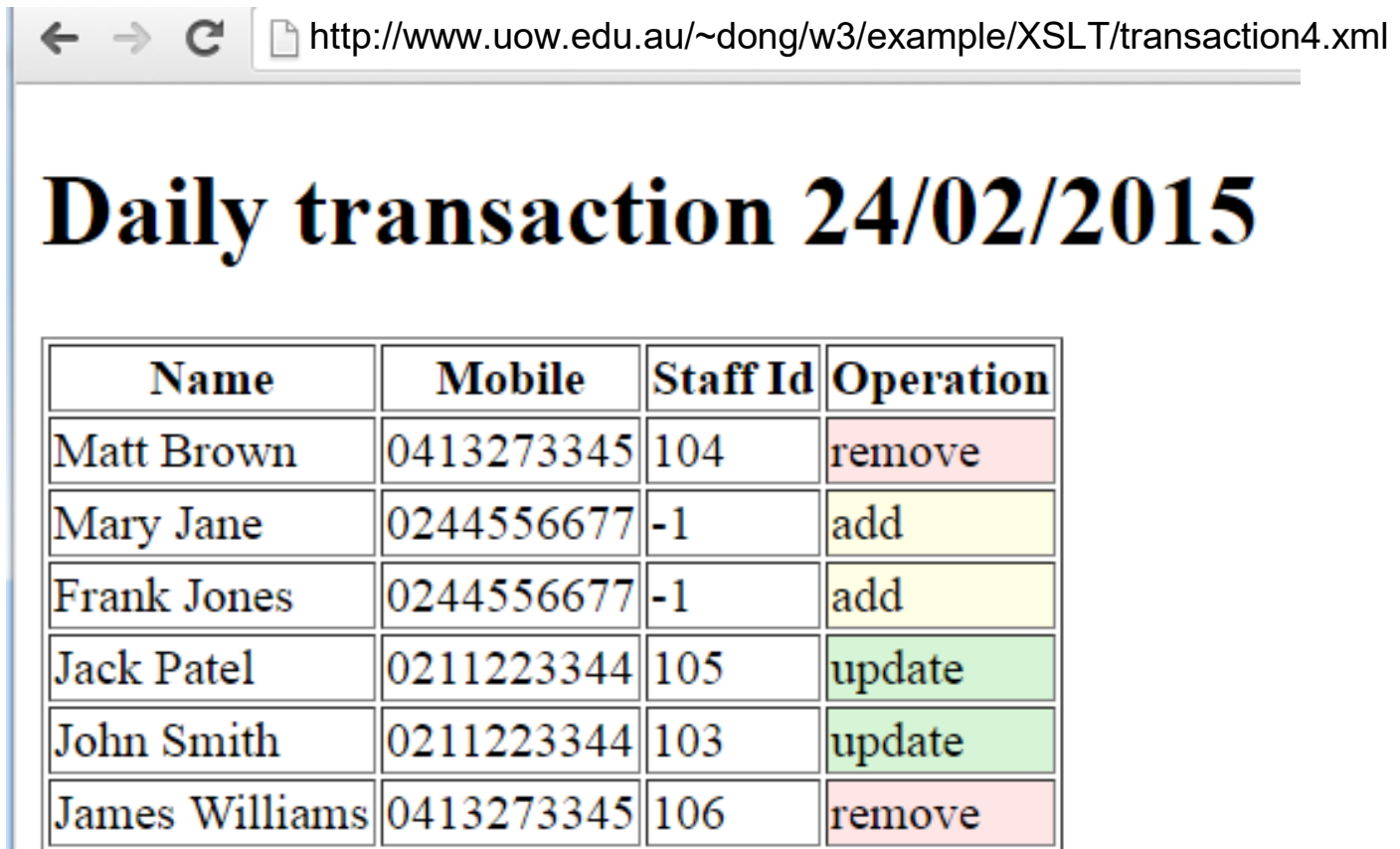
← → ↻ www.uow.edu.au/~dong/ISIT206/XSLT/trans

Daily transaction 24/02/2006

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

XSLT

Example 4: Now look at `transaction4.xml`



← → ↻ <http://www.uow.edu.au/~dong/w3/example/XSLT/transaction4.xml>

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

XSLT

Now look at transaction-style4.xsl

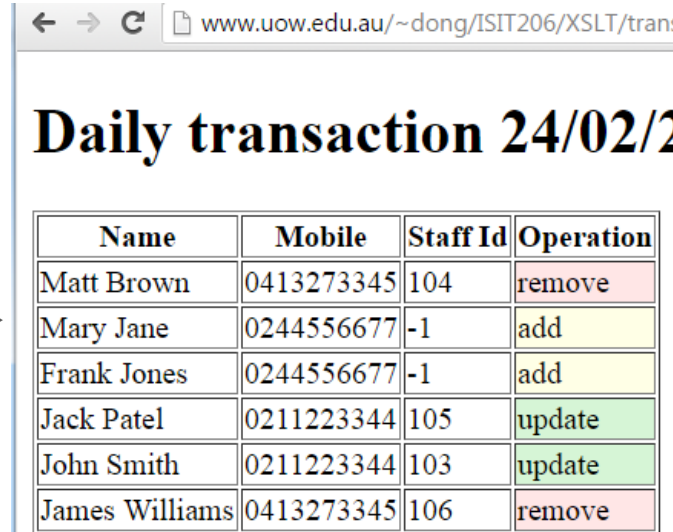
```
<xsl:choose>
```

```
  <xsl:when test="@operation = 'remove'">
    <td bgcolor="#ffe6e6">
      <xsl:value-of select="@operation" />
    </td>
  </xsl:when>
```

```
  <xsl:when test="@operation = 'add'">
    <td bgcolor="#ffffe6">
      <xsl:value-of select="@operation" />
    </td>
  </xsl:when>
```

```
  <xsl:otherwise>
    <td bgcolor="#d6f5d6">
      <xsl:value-of select="@operation" />
    </td>
  </xsl:otherwise>
```

```
</xsl:choose>
```



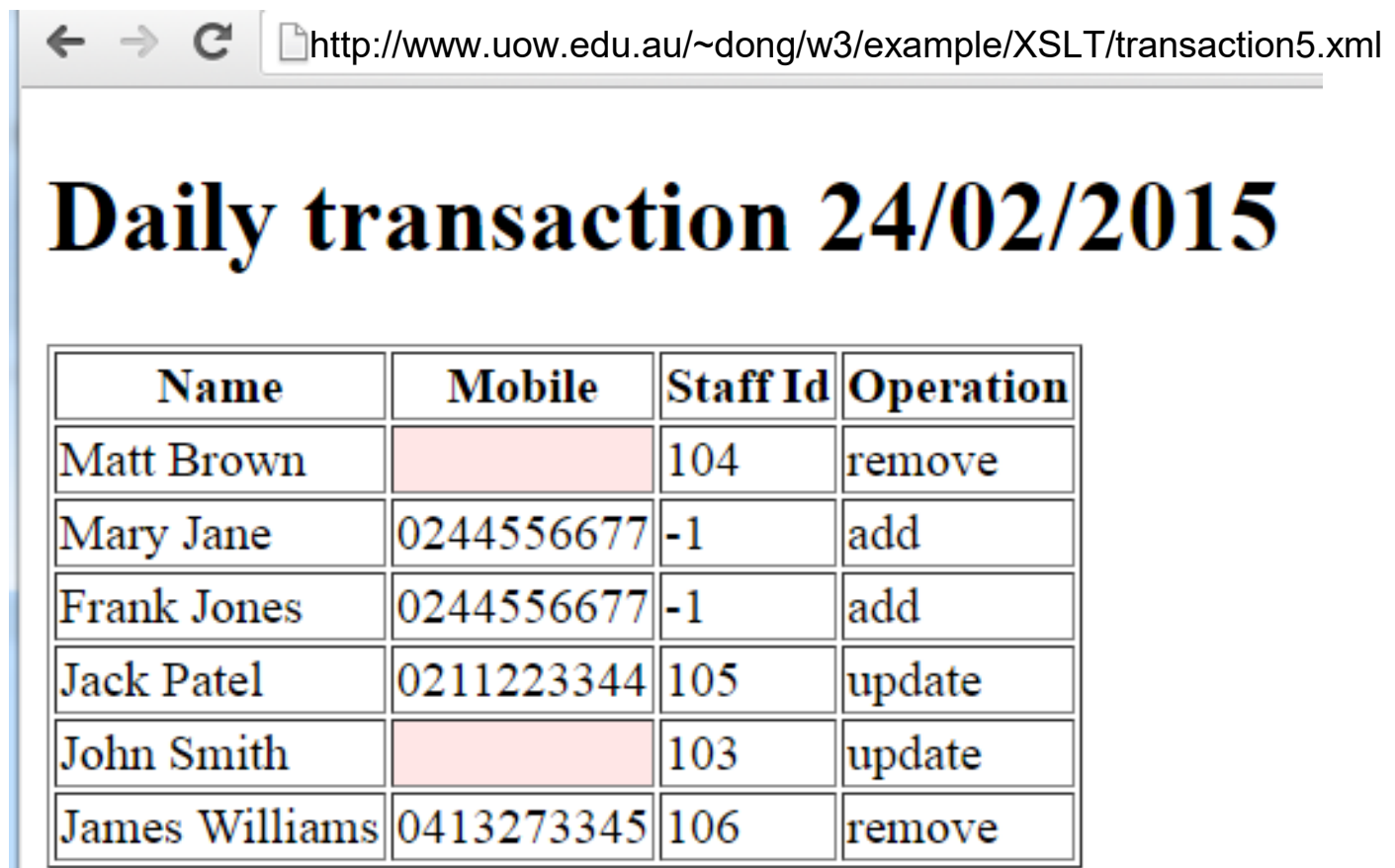
← → ↻ www.uow.edu.au/~dong/ISIT206/XSLT/trans

Daily transaction 24/02/2020

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

XSLT

Example 5: Now look at `transaction5.xml`



← → ↻ <http://www.uow.edu.au/~dong/w3/example/XSLT/transaction5.xml>

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown		104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith		103	update
James Williams	0413273345	106	remove

XSLT

Now look at transaction-style5.xsl

```
<xsl:choose>
```

```
<xsl:when test="mobile = ''">  
  <td bgcolor="#ffe6e6"> </td>  
</xsl:when>
```

```
<xsl:otherwise>  
  <td> <xsl:value-of select="mobile" /> </td>  
</xsl:otherwise>
```

```
</xsl:choose>
```

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown		104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith		103	update
James Williams	0413273345	106	remove

XSLT

Example 6: Now look at `transaction6.xml`

← → ↻  <http://www.uow.edu.au/~dong/w3/example/XSLT/transaction6.xml>

Daily transaction 24/02/2015

New records

Name	Mobile	Staff Id	Operation
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add

Updated records

Name	Mobile	Staff Id	Operation
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update

Removed records

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
James Williams	0413273345	106	remove

XSLT

Now look at transaction-style6.xsl

```
<xsl:for-each select="person[@operation='add']">
  <xsl:sort select="lastName"/>
  <tr>
    <td>
      <xsl:value-of select="firstName" />
      <xsl:text> </xsl:text>
      <xsl:value-of select="lastName" />
    </td>
    <td>
      <xsl:value-of select="mobile" />
    </td>
    <td>
      <xsl:value-of select="@staffDbId" />
    </td>
    <td>
      <xsl:value-of select="@operation" />
    </td>
  </tr>
</xsl:for-each>
```

← → ↻ www.uow.edu.au/~dong/ISIT206/XSLT/transaction-style6.html

Daily transaction 24/02/20

New records

Name	Mobile	Staff Id	Operation
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add

Updated records

Name	Mobile	Staff Id	Operation
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update

Removed records







Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
James Williams	0413273345	106	remove

XSLT

Example 7: Now look at `transaction7.xml`

<http://www.uow.edu.au/~dong/w3/example/XSLT/transaction7.xml>

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	
Mary Jane	0244556677	-1	
Frank Jones	0244556677	-1	
Jack Patel	0211223344	105	
John Smith	0211223344	103	
James Williams	0413273345	106	

XSLT


Now look at transaction-style7.xsl

```
...
<td>
  <xsl:value-of select="@staffDbId" />
</td>

<td align="center">
  <img>
    <xsl:attribute name="src">
      <xsl:text>images/</xsl:text>
      <xsl:value-of select="@operation"/>
      <xsl:text>.png</xsl:text>
    </xsl:attribute>

    <xsl:attribute name="width">
      <xsl:text>30px</xsl:text>
    </xsl:attribute>
  </img>
</td>
...
```

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	
Mary Jane	0244556677	-1	
Frank Jones	0244556677	-1	
Jack Patel	0211223344	105	
John Smith	0211223344	103	
James Williams	0413273345	106	

XSLT

Example 8: Now look at `transaction8.xml`

<http://www.uow.edu.au/~dong/w3/example/XSLT/transaction8.xml>

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

XSLT

Now look at transaction-style8.xsl

```
...
<td>
  <xsl:choose>
    <xsl:when test="number(@staffDbId) < 0">
      <span style="color:red">
        <xsl:value-of select="@staffDbId" />
      </span>
    </xsl:when>

    <xsl:otherwise>
      <span style="color:green">
        <xsl:value-of select="@staffDbId" />
      </span>
    </xsl:otherwise>
  </xsl:choose>
</td>
...
```

Daily transaction 24/02/2015

Name	Mobile	Staff Id	Operation
Matt Brown	0413273345	104	remove
Mary Jane	0244556677	-1	add
Frank Jones	0244556677	-1	add
Jack Patel	0211223344	105	update
John Smith	0211223344	103	update
James Williams	0413273345	106	remove

References

`http://www.w3schools.com/xsl`

Robert W. Sebesta, *Programming the World Wide Web*, Pearson.