

XSL Transformations

Šta je XSLT

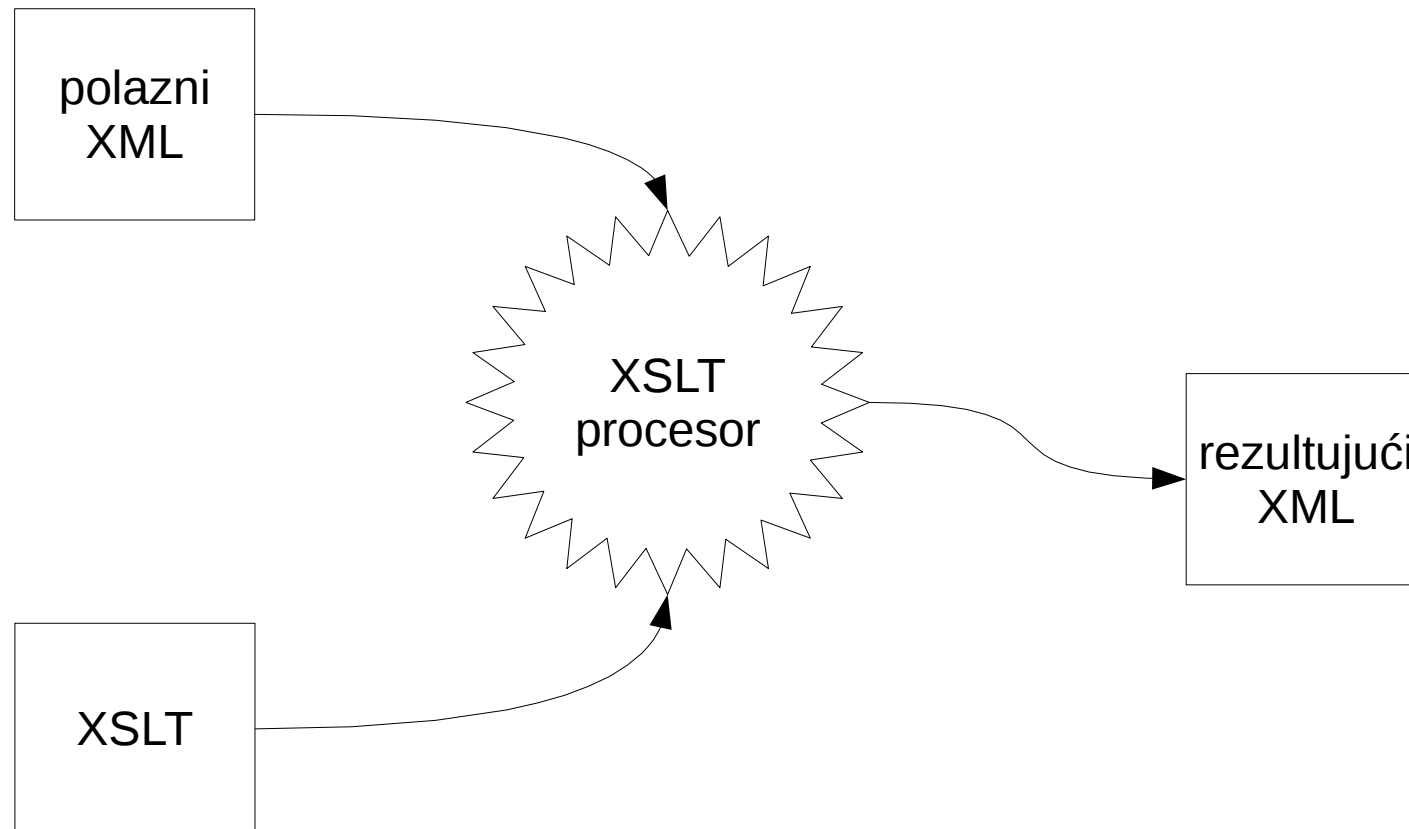
- jezik za opis transformacija XML dokumenata u druge XML dokumente
- W3C standard
- za navigaciju koristi XPath izraze
- vizuelizacija - samo jedna od primena
 - kada je rezultat transformacije XHTML

Šta je XSLT

- XSLT je još jedna XML gramatika
- namespace je `http://www.w3.org/1999/XSL/Transform`

Kako radi XSLT

- jedan XML dokument se transformiše u drugi
- transformacija je opisana XSLT dokumentom
- transformaciju izvodi XSLT procesor



Kako radi XSLT

- XSLT fajl sadrži definicije šablona
 - šablon opisuje transformaciju dela polaznog dokumenta
 - deo polaznog dokumenta se identifikuje XPath izrazom
- interpretiranje XSLT fajla se svodi na primenu šablona

Primer XSLT transformacije

- polazni XML dokument koji se transformiše

```
<?xml version="1.0"?>
<catalog>
  <cd>
    <title>Empire Burlesque</title>
    <artist>Bob Dylan</artist>
    <country>USA</country>
    <company>Columbia</company>
    <price>10.90</price>
    <year>1985</year>
  </cd>
  ...
</catalog>
```

Primer XSLT transformacije

- XSLT transformacija

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

koren svakog
XSLT fajla

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

```
  <html>
```

```
  <body>
```

```
    ...
```

```
  </body>
```

```
  </html>
```

```
</xsl:template>
```

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

šablon se primenjuje na
korenski element (XPath
izraz je /)

kada se šablon primeni,
njegov sadržaj se dodaje na
kraj rezultujućeg fajla;
ovde je rezultujući fajl u
HTML formatu

Primer XSLT transformacije

- rezultat transformacije

```
<html>  
<body>  
...  
</body>  
</html>
```


Primer XSLT transformacije

- sadržaj templejta

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

Šablon se primenjuje na koren;
koren postaje tekući kontekst

```
<html>
```

```
<body>
```

```
<h2>My CD Collection</h2>
```

```
<table border="1">
```

```
<tr bgcolor="#9acd32">
```

```
<th align="left">Title</th>
```

```
<th align="left">Artist</th>
```

```
</tr>
```

```
<xsl:for-each select="catalog/cd">
```

```
<tr>
```

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

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

```
</tr>
```

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

```
</table>
```

```
</body>
```

```
</html>
```

```
</xsl:template>
```

Elementi catalog/cd se
traže u okviru tekućeg
konteksta, a to je /.
Za svaki pronađeni <cd>
ponavlja se sledeći blok

Ovde se u sadržaj ubacuje vrednost elemenata
title i artist. Oni se traže u tekućem kontekstu,
a to je cd

HTML tagovi su plavi
XSLT tagovi su crveni

Primer XSLT transformacije

- rezultat transformacije

```
<html>
<body>
  <h2>My CD Collection</h2>
  <table border="1">
    <tr bgcolor="#9acd32">
      <th align="left">Title</th>
      <th align="left">Artist</th>
    </tr>
    <tr>
      <td>Empire Burlesque</td>
      <td>Bob Dylan</td>
    </tr>
  </table>
</body>
</html>
```

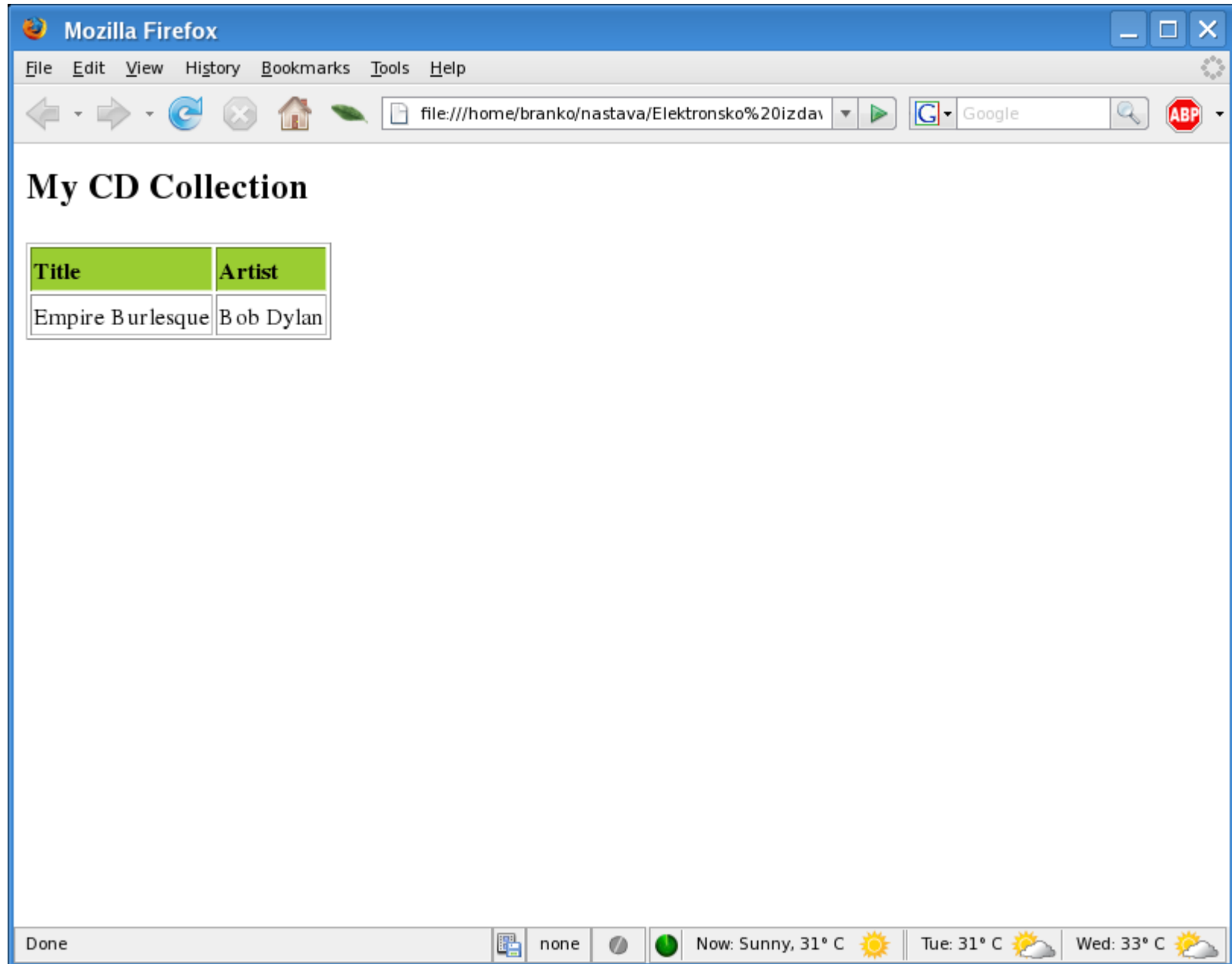
ovaj sadržaj je direktno
prepisan iz šablona

ovaj deo se ponavlja onoliko
puta koliko ima cd elemenata,
sa njihovim odgovarajućim
sadržajem

ovaj sadržaj je
ubačen iz XML
dokumenta

Primer XSLT transformacije

- rezultat transformacije - prikaz u web browseru



Povezivanje dokumenta sa prikazom

- koristi se `xml-stylesheet` procesna instrukcija

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="cdcatalog.xsl"?>
<catalog>
  <cd>
    <title>Empire Burlesque</title>
    <artist>Bob Dylan</artist>
    <country>USA</country>
    <company>Columbia</company>
    <price>10.90</price>
    <year>1985</year>
  </cd>
  ...
</catalog>
```

<xsl:stylesheet>

- korenski element svakog XSLT dokumenta
 - atribut `version` - verzija XSLT jezika koja se koristi u fajlu
 - deklaracija za XSLT namespace, uobičajeni prefiks je `xsl`
 - sadržaj elementa čine
 - šabloni: `<xsl:template>`

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

<xsl:template>

- definiše šablon
 - atribut `match` sadrži kontekst u kome će se šablon izvršiti
 - kontekst se definiše kao XPath izraz
 - XPath izraz, ako predstavlja relativnu putanju, se izračunava u odnosu na tekući kontekst
 - ako se ne pronađe dati kontekst, šablon se ne izvršava
 - sadržaj elementa se dodaje na kraj rezultujućeg fajla
 - ako se u sadržaju nalaze drugi XSLT elementi, oni se prethodno izračunavaju

```
<xsl:template match="/catalog/cd">
```

```
...
```

```
</xsl:template>
```

<xsl:value-of>

- sadržaj datog elementa dodaje u rezultujući dokument
 - atribut `select` predstavlja XPath putanju do željenog elementa
 - ako element ne postoji, ne ubacuje se ništa
 - ako postoji više elemenata za dati XPath, uzima se prvi
 - element `<xsl:value-of>` je uvek prazan

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

<xsl:for-each>

- izdvaja svaki element iz skupa čvorova
 - atribut `select` predstavlja XPath putanju do željenih elemenata
 - sadržaj `<xsl:for-each>` se dodaje na izlazni fajl za svaki pronađeni element
 - sadržaj se ponovo procesira ako sadrži XSLT elemente

```
<xsl:for-each select="catalog/cd">
```

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

```
<xsl:for-each select="catalog/cd[artist='Bob Dylan']">
```

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



filtriranje podataka

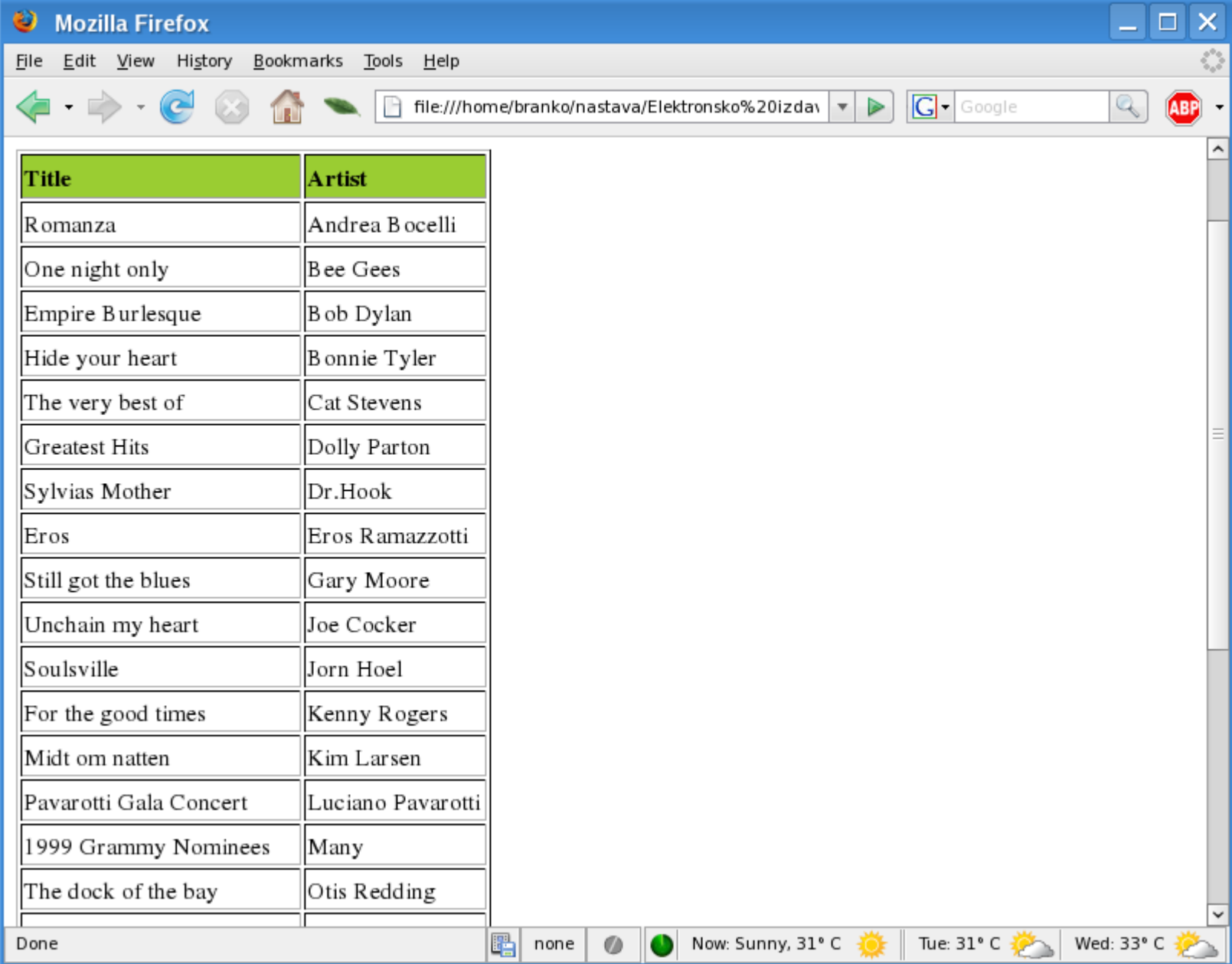
<xsl:sort>

- sortira sadržaj tekućeg konteksta prema datom kriterijumu
 - atributi
 - select - XPath do čvora čiji sadržaj se koristi kao ključ za sortiranje
 - data-type - koji tip podatka se koristi kao ključ (text, number, QName)
 - order - opadajući ili rastući redosled (ascending, descending)
 - case-order - prvo velika ili mala slova (upper-first, lower-first)
 - lang - jezik čiji alfabet se koristi za sortiranje (ISO 639: sr, en, fr, itd)
 - element je uvek prazan

```
<xsl:for-each select="catalog/cd">  
  <xsl:sort select="artist"/>  
  ...  
</xsl:for-each>
```

<xsl:sort>

- primer sortiranog dokumenta



The screenshot shows a Mozilla Firefox browser window. The address bar displays the file path: `file:///home/branko/nastava/Elektronsko%20izdav`. The browser is displaying a table with two columns: **Title** and **Artist**. The table contains 16 rows of music data. The status bar at the bottom shows the system clock and weather information for the current location.

Title	Artist
Romanza	Andrea Bocelli
One night only	Bee Gees
Empire Burlesque	Bob Dylan
Hide your heart	Bonnie Tyler
The very best of	Cat Stevens
Greatest Hits	Dolly Parton
Sylvias Mother	Dr.Hook
Eros	Eros Ramazzotti
Still got the blues	Gary Moore
Unchain my heart	Joe Cocker
Soulsville	Jorn Hoel
For the good times	Kenny Rogers
Midt om natten	Kim Larsen
Pavarotti Gala Concert	Luciano Pavarotti
1999 Grammy Nominees	Many
The dock of the bay	Otis Redding

<xsl:if>

- prikazuje svoj sadržaj samo ako je dati uslov ispunjen
 - atribut test - uslov koji je potrebno ispuniti
 -

```
<xsl:for-each select="catalog/cd">
  <xsl:if test="price < 10"/>
    <tr>
      <td><xsl:value-of select="title"/></td>
      <td><xsl:value-of select="artist"/></td>
    </tr>
  </xsl:if>
</xsl:for-each>
```

znak < nije dozvoljen
pa se mora pisati <

dodaće se u
rezultat samo
ako je price < 10

<xsl:choose>

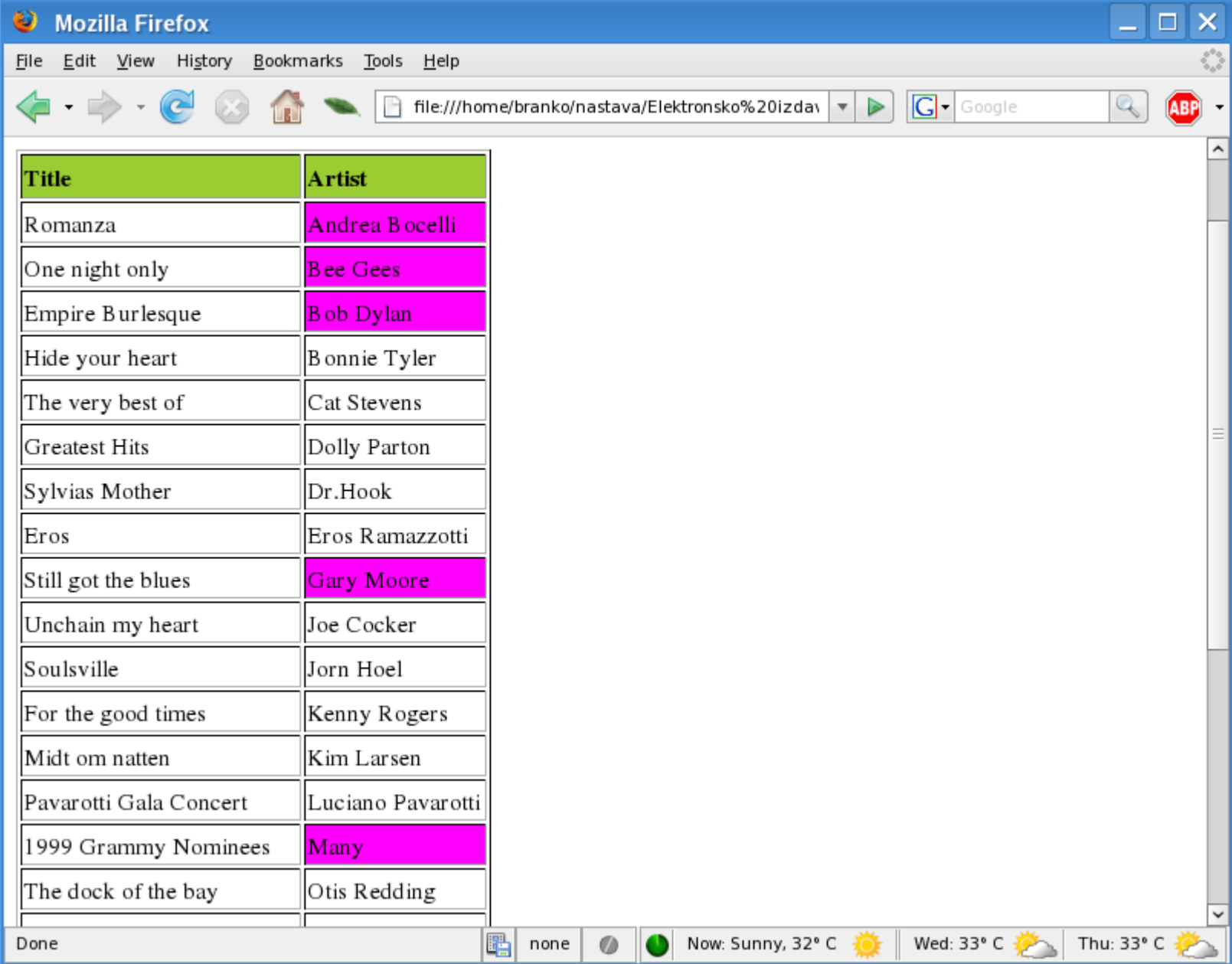
- izvršava više uzastopnih testova
- svaki test se opisuje pomoću <xsl:when> podelementa
 - atribut test predstavlja sadržaj testa
- ako nijedan test nije ispunjen, izvršava se <xsl:otherwise> podelement

-

```
<xsl:choose>
  <xsl:when test="price > 10">
    <td bgcolor="#ff00ff">
      <xsl:value-of select="artist"/></td>
    </xsl:when>
    <xsl:otherwise>
      <td><xsl:value-of select="artist"/></td>
    </xsl:otherwise>
  </xsl:choose>
```

<xsl:choose>

- primer različito obojenih redova u tabeli



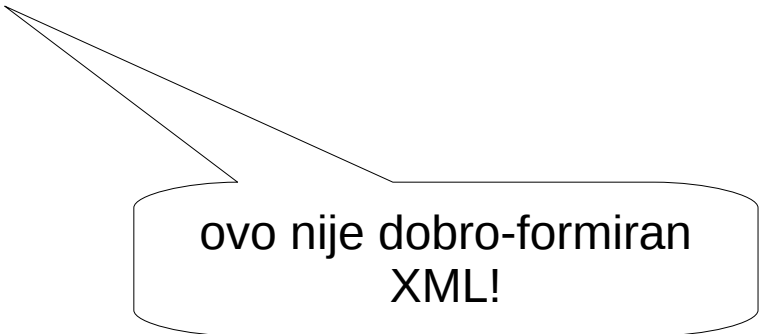
The screenshot shows a Mozilla Firefox browser window displaying a table. The table has two columns: 'Title' and 'Artist'. The rows are color-coded: the first row has a green header, and subsequent rows alternate between white and magenta backgrounds. The browser's address bar shows a local file path, and the status bar at the bottom displays weather information.

Title	Artist
Romanza	Andrea Bocelli
One night only	Bee Gees
Empire Burlesque	Bob Dylan
Hide your heart	Bonnie Tyler
The very best of	Cat Stevens
Greatest Hits	Dolly Parton
Sylvias Mother	Dr.Hook
Eros	Eros Ramazzotti
Still got the blues	Gary Moore
Unchain my heart	Joe Cocker
Soulsville	Jorn Hoel
For the good times	Kenny Rogers
Midt om natten	Kim Larsen
Pavarotti Gala Concert	Luciano Pavarotti
1999 Grammy Nominees	Many
The dock of the bay	Otis Redding

<xsl:choose>

- da li je prethodni primer mogao da se zapiše kraće, pomoću xsl:if?

```
<td <xsl:if test="price < 10">bgcolor="#ff00ff"</xsl:if>  
  <xsl:value-of select="artist"/>  
</td>
```



ovo nije dobro-formiran
XML!

<xsl:apply-templates>

- pokreće ponovnu primenu šablona u okviru tekućeg konteksta
 - ako ima atribut `select` šabloni se primenjuju samo na izabrane podčvorove

```
<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: <xsl:value-of select="."/><br/>
</xsl:template>
```

<xsl:call-template>

- poziva imenovani šablon na datom mestu

```
<xsl:template name="createHeader">
```

```
...
```

```
</xsl:template>
```

```
...
```

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

```
  <html>
```

```
    <head>
```

```
      <title>
```

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

```
      </title>
```

```
    </head>
```

```
  <body>
```

```
    <xsl:call-template name="createHeader"/>
```


Parametri

- šablon može da ima parametre

```
<xsl:template name="calcuatArea">  
  <xsl:param name="width"/>  
  <xsl:param name="height"/>
```

parametri se moraju
definisati na početku
šablona

```
  <xsl:value-of select="$width * $height"/>  
</xsl:template>
```

vrednost parametra se
poziva sa znakom \$ ispred

Parametri

- parametri mogu imati podrazumevane vrednosti

```
<template name="addTableCell">
  <xsl:param name="bgColor" select="'blue'"/>
  <xsl:param name="width" select="150"/>
  <xsl:param name="content"/>

  <td width="{ $width}" bgcolor="{ $bgColor}">
    <xsl:apply-templates select="$content"/>
  </td>
</template>
```

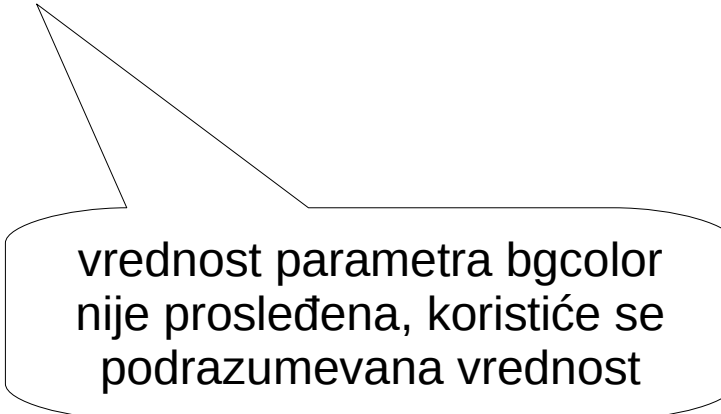
da piše blue mislilo
bi se na podelement
u tekućem kontekstu

upotreba
parametara u tekstu:
{ \$param }

Parametri

- prosleđivanje vrednosti parametara prilikom poziva šablona

```
<xsl:call-template name="addTableCell">  
  <xsl:with-param name="width" select="100"/>  
  <xsl:with-param name="content" select="$title">  
</xsl:call-template>
```



vrednost parametra bgcolor
nije prosleđena, koristiće se
podrazumevana vrednost

Parametri

- inicijalizacija parametra može biti i unutar `<xsl:param>` elementa

```
<xsl:param name="bgColor">blue</xsl:param>
```

```
<xsl:param name="width">  
  <xsl:value-of select="7+8"/>  
  <xsl:text>0</xsl:text>  
</xsl:param>
```

Promenljive

- definisanje promenljive

```
<xsl:variable name="x"/>
```

```
<xsl:variable name="x" select="'1'"/>
```

```
<xsl:variable name="y">  
  <xsl:choose>  
    <xsl:when test="$x > 7">  
      <xsl:text>13</xsl:text>  
    </xsl:when>  
    <xsl:otherwise>  
      <xsl:text>15</xsl:text>  
    </xsl:otherwise>  
  </xsl:choose>  
</xsl:variable>
```

Promenljive

- upotreba promenljive - isto kao i kod parametra

`$promenljiva`

Promenljive

- promena vrednosti promenljive
 - nije moguća
- promenljive su vezane za šablon u kojima su definisane
 - različiti šabloni mogu imati promenljive istog imena
- globalna promenljiva - dete `<xsl:stylesheet>` elementa
 - vidljiva u svim šablonima koji se nalaze posle njene definicije

Primer složenije vizuelizacije

- XML dokument

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl"
  href="daltons.xsl"?>
<family>
  <member>
    <name>
      <first>Joe</first>
      <last>Dalton</last>
    </name>
    <iq>100</iq>
    <height>160</height>
  </member>
  <member>
    <name>
      <first>William</first>
      <last>Dalton</last>
    </name>
    <iq>95</iq>
    <height>170</height>
  </member>
```

```
<member>
  <name>
    <first>Jack</first>
    <last>Dalton</last>
  </name>
  <iq>90</iq>
  <height>180</height>
</member>
<member>
  <name>
    <first>Averel</first>
    <last>Dalton</last>
  </name>
  <iq>85</iq>
  <height>190</height>
</member>
</family>
```


Primer složenije vizuelizacije

- XSL transformacija

```
<xsl:template match="/">
<html>
  <body>
    <table border="1">
      <tr>
        <th>Ime</th>
        <th>Prezime</th>
        <th>Visina</th>
        <th>IQ</th>
      </tr>
      <xsl:for-each select="family/member">
        <tr>
          <td><xsl:value-of select="name/first"/></td>
          <td><xsl:value-of select="name/last"/></td>
          <td><xsl:value-of select="height"/></td>
          <td><xsl:value-of select="iq"/></td>
        </tr>
      </xsl:for-each>
    </table>
  </body>
</html>
</xsl:template>
```

Primer složenije vizuelizacije

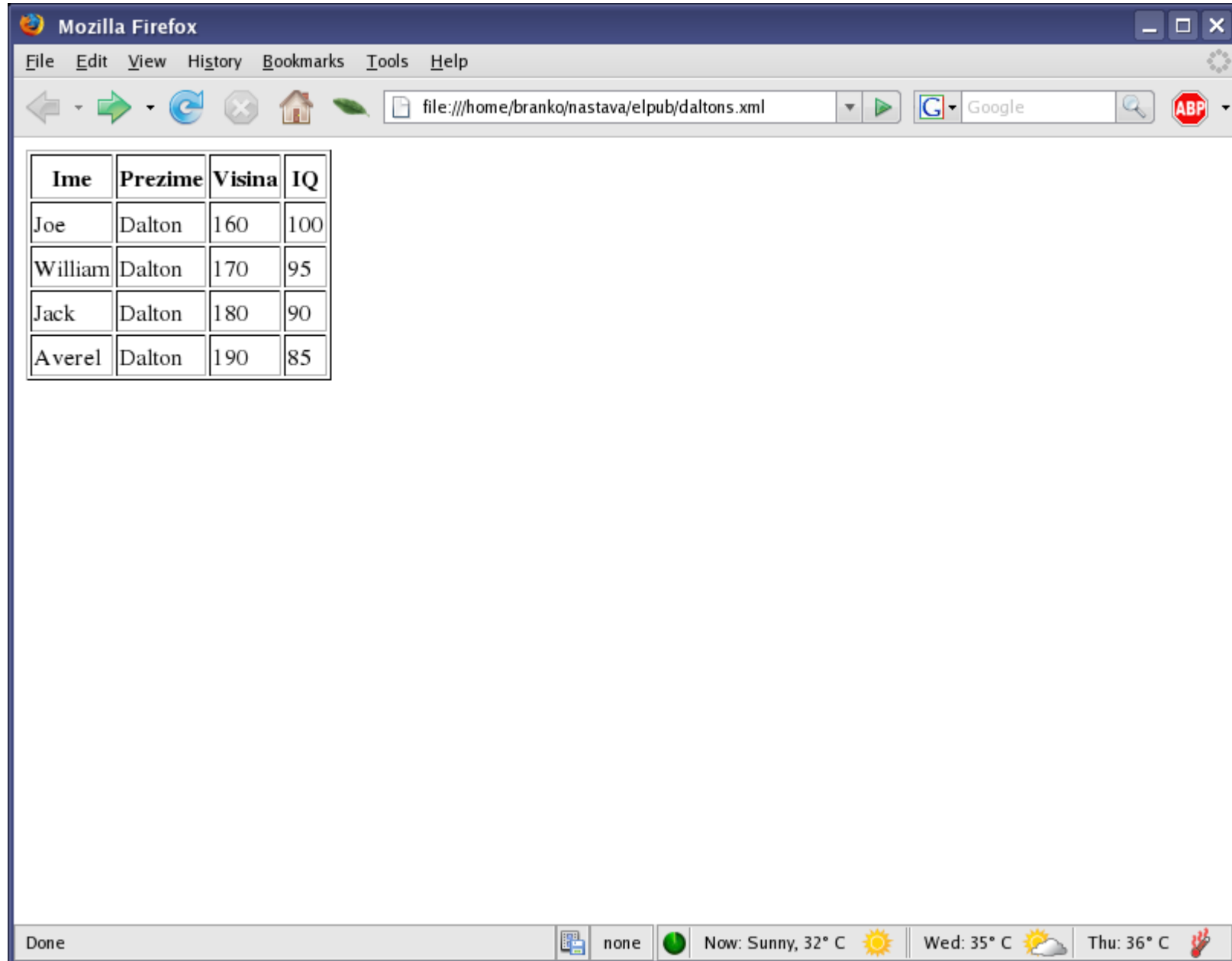
- dobijeni rezultat

```
<html>
<body>
<table border="1">
  <tr>
    <th>Ime</th>
    <th>Prezime</th>
    <th>Visina</th>
    <th>IQ</th>
  </tr>
  <tr>
    <td>Joe</td>
    <td>Dalton</td>
    <td>160</td>
    <td>100</td>
  </tr>
  <tr>
    <td>William</td>
    <td>Dalton</td>
    <td>170</td>
    <td>95</td>
  </tr>
```

```
<tr>
  <td>Jack</td>
  <td>Dalton</td>
  <td>180</td>
  <td>90</td>
</tr>
<tr>
  <td>Averel</td>
  <td>Dalton</td>
  <td>190</td>
  <td>85</td>
</tr>
</table>
</body>
</html>
```

Primer složenije vizuelizacije

- izgled rezultata u browseru



Primer složenije vizuelizacije

- računanje proseka za tabelu

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

```
...
```

```
<tr>
```

```
  <td colspan="2">PROSEK</td>
```

```
  <xsl:variable name="sumheight"
```

```
    select="sum(family/member/height)"/>
```

```
  <xsl:variable name="sumiq"
```

```
    select="sum(family/member/iq)"/>
```

```
  <xsl:variable name="total"
```

```
    select="count(family/member)"/>
```

```
  <td><xsl:value-of select="$sumheight div $total"/></td>
```

```
  <td><xsl:value-of select="$sumiq div $total"/></td>
```

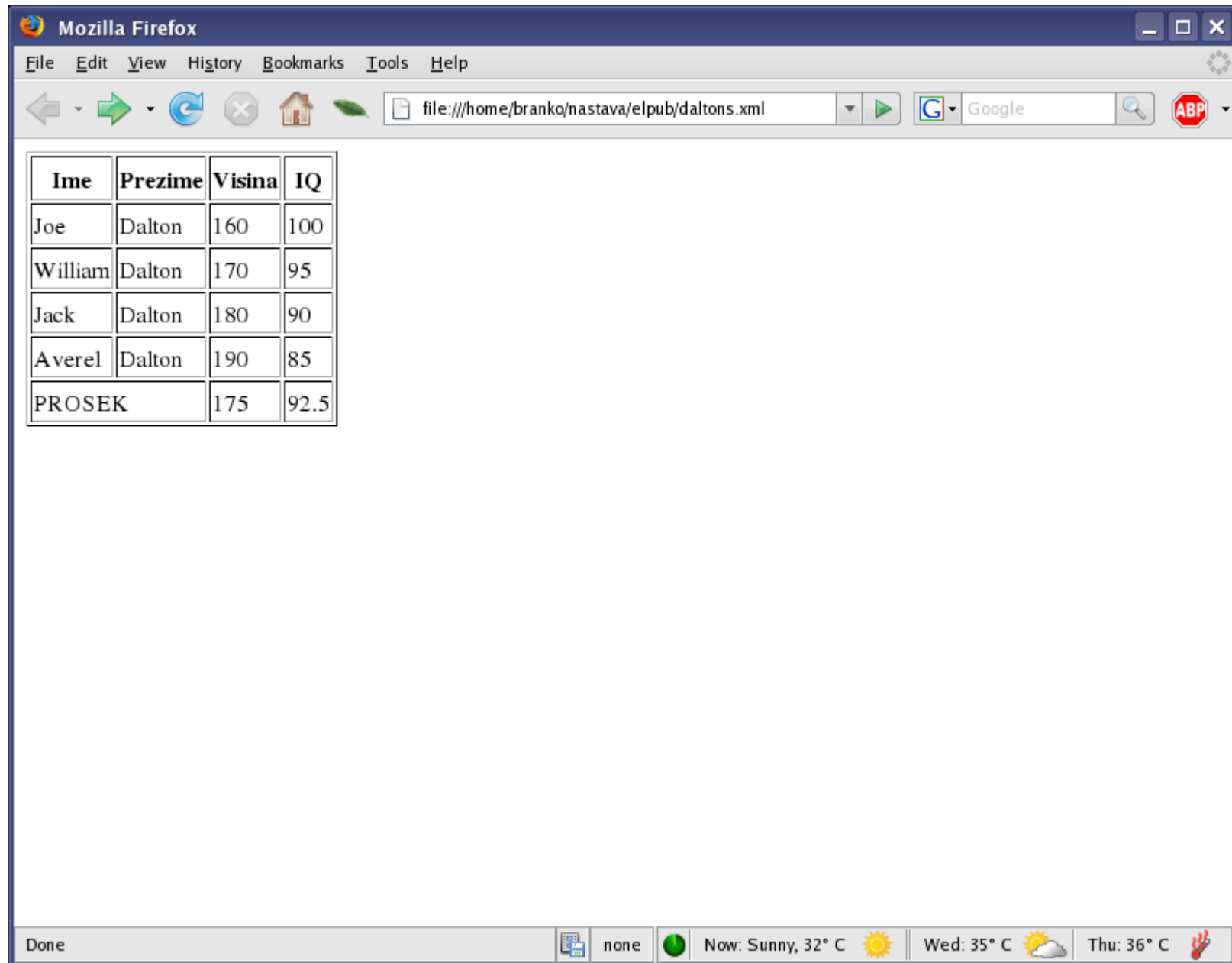
```
</tr>
```

```
...
```

```
</xsl:template>
```

Primer složenije vizuelizacije

- izgled rezultata u browseru



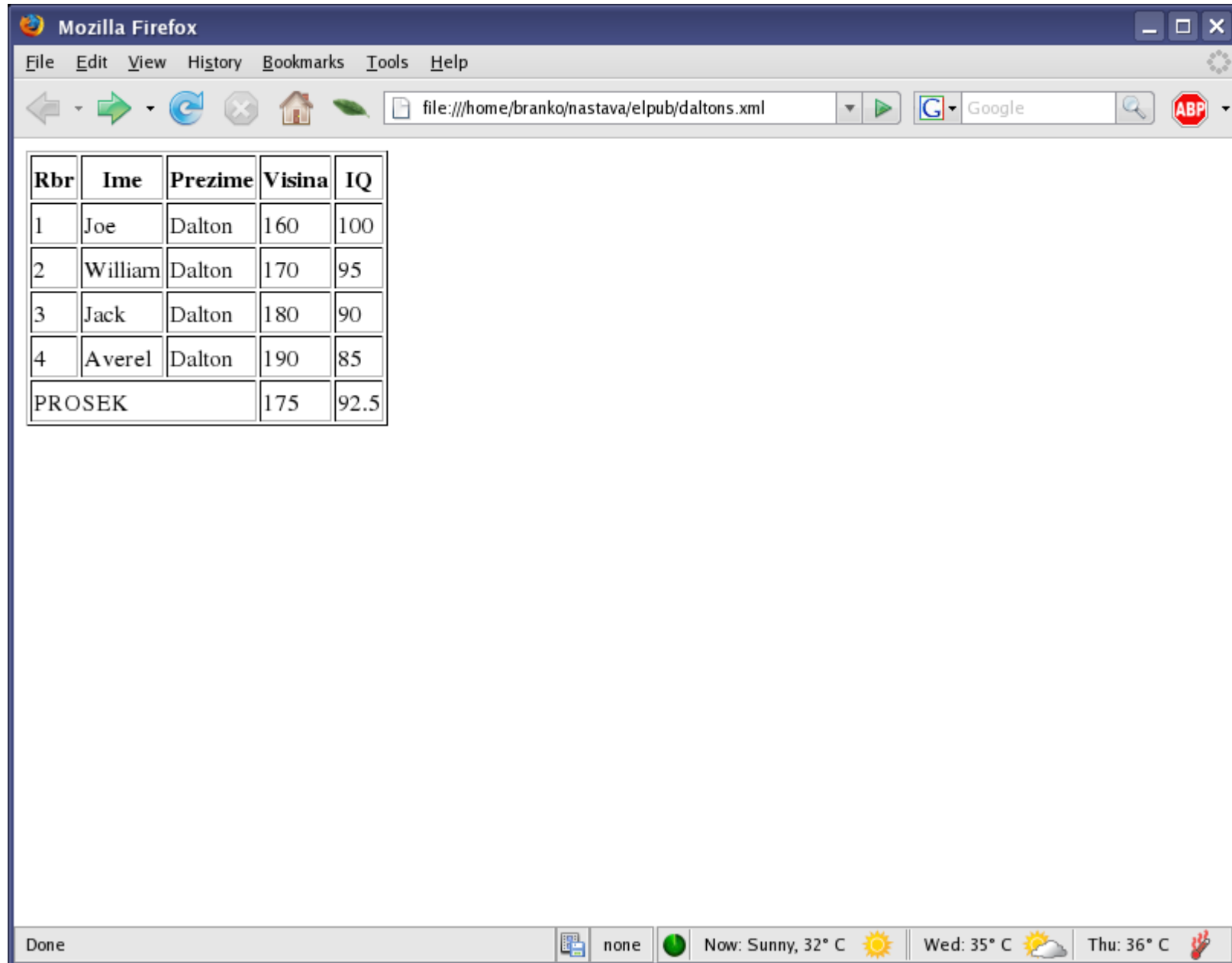
Primer složenije vizuelizacije

- dodavanje rednog broja u tabelu

```
<xsl:for-each select="family/member">
  <tr>
    <td><xsl:number/></td>
    <td><xsl:value-of select="name/first"/></td>
    <td><xsl:value-of select="name/last"/></td>
    <td><xsl:value-of select="height"/></td>
    <td><xsl:value-of select="iq"/></td>
  </tr>
</xsl:for-each>
```

Primer složenije vizuelizacije

- izgled rezultata u browseru



The screenshot shows a Mozilla Firefox browser window. The address bar displays the file path: `file:///home/branko/nastava/elpub/daltons.xml`. The browser has rendered an XML document as a table with the following data:

Rbr	Ime	Prezime	Visina	IQ
1	Joe	Dalton	160	100
2	William	Dalton	170	95
3	Jack	Dalton	180	90
4	Averel	Dalton	190	85
PROSEK			175	92.5

The browser's status bar at the bottom shows "Done" and a weather forecast: "Now: Sunny, 32° C", "Wed: 35° C", and "Thu: 36° C".

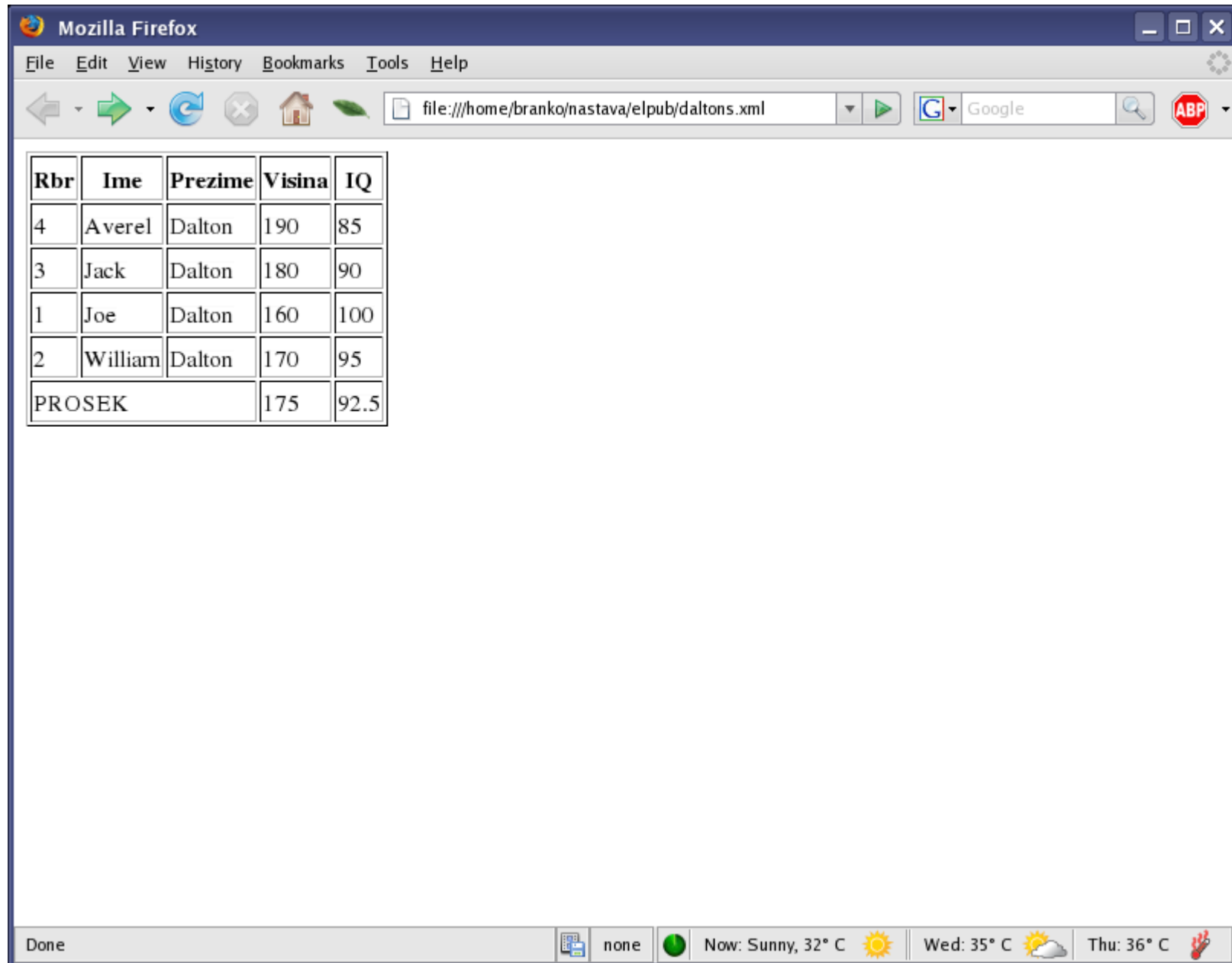
Primer složenije vizuelizacije

- sortiranje po imenu

```
<xsl:for-each select="family/member">
  <xsl:sort select="name/first"/>
  <tr>
    <td><xsl:number/></td>
    <td><xsl:value-of select="name/first"/></td>
    <td><xsl:value-of select="name/last"/></td>
    <td><xsl:value-of select="height"/></td>
    <td><xsl:value-of select="iq"/></td>
  </tr>
</xsl:for-each>
```


Primer složenije vizuelizacije

- izgled rezultata u browseru



Primer složenije vizuelizacije

- sortiranje i numeracija

```
<xsl:for-each select="family/member">
  <xsl:sort select="name/first"/>
  <tr>
    <td><xsl:value-of select="position()"/></td>
    <td><xsl:value-of select="name/first"/></td>
    <td><xsl:value-of select="name/last"/></td>
    <td><xsl:value-of select="height"/></td>
    <td><xsl:value-of select="iq"/></td>
  </tr>
</xsl:for-each>
```

<xsl:number> uvek numeriče čvorove prema njihovom položaju u originalnom stablu dokumenta

XPath funkcija position() vraća redni broj čvora u tekućem kontekstu, ali se pri tome uzima u obzir sortirani kontekst