CSC343 Worksheet 10 Solution

June 28, 2020

1. a) **Notes:**
   - XPATH and Selecting Nodes
     - nodename
       - Selects all nodes with the name "nodename"
     - /
       - Selects from the root node
     - //
       - Selects node in the document from the current node that match the selection no matter where they are
     - .
       - Select the current node
     - ..
       - Selects the parent of the current node
     - @
       - Selects attributes

     Example:

     ```
     /StarMovieData/Star//City
     ```

     - selects all City element in

     <StarMovieData>
       <Star>
         Here :)
       </Star>
     </StarMovieData>
   - Wildcards *
     - Is used to say 'any tag'

Example:

```
/StarMovieData/*/@*
```

  - '@*' means any attributes
  - '*' means any tag

- Context of Expressions
  - [...] means that exists or there exists
  - [*integer*] selects ith child of its parent
  - [*Tag*] selects elements that have one or more sublements with 'Tag'
  - [*Attribute*] selects elements that have attribute 'Attribute'

Example:

```
/StarMovieData/Star[//City = "Malibu"]/Name
```

∗ Means select all Star Name that contains City with value 'Malibu'

Example:

```
/Movies/Movie/Version[1]/@year
```

∗ Returns value of 'year' attribute of first 'Version' tag in 'Movie'

∗ e.g. 1933 and 1984

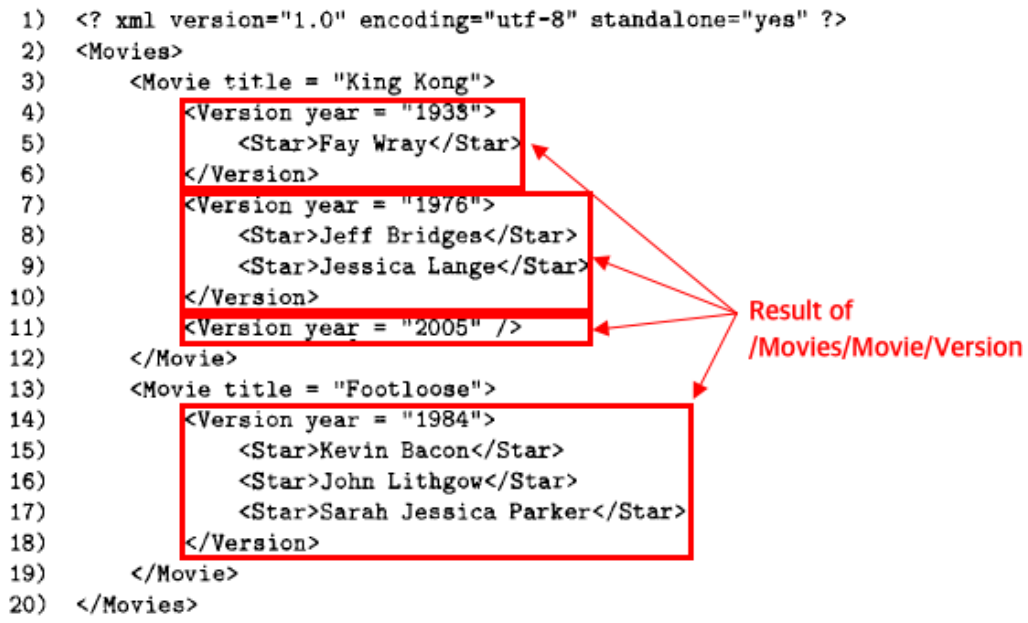
```
1)  <? xml version="1.0" encoding="utf-8" standalone="yes" ?>
2)  <Movies>
3)      <Movie title = "King Kong">
4)          <Version year = "1933">
5)              <Star>Fay Wray</Star>
6)          </Version>
7)          <Version year = "1976">
8)              <Star>Jeff Bridges</Star>
9)              <Star>Jessica Lange</Star>
10)         </Version>
11)         <Version year = "2005" />
12)     </Movie>
13)     <Movie title = "Footloose">
14)         <Version year = "1984">
15)             <Star>Kevin Bacon</Star>
16)             <Star>John Lithgow</Star>
17)             <Star>Sarah Jessica Parker</Star>
18)         </Version>
19)     </Movie>
20) </Movies>
```

Result of /Movies/Movie/Version[1]/@year

Example 2:

```
1    /Movies/Movie/Version
2
```

* Returns all 'Version' tag in 'Movie'
* e.g. lines 4 through 6, 7 through 10, line 11, lines 14 through 18

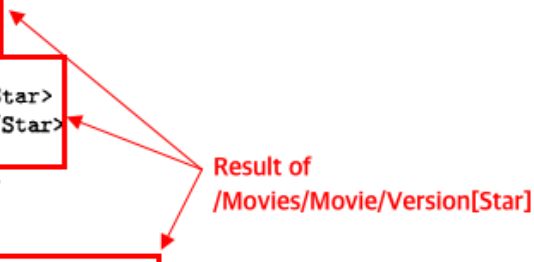


Example 3:

```
1    /Movies/Movie/Version[Star]
2
```

* Selects all 'Version' tag with one or more 'Star' tag inside
* e.g lines 4 through 6, 7 through 10, 14 through 18

```
1)  <? xml version="1.0" encoding="utf-8" standalone="yes" ?>
2)  <Movies>
3)      <Movie title = "King Kong">
4)          <Version year = "1933">
5)              <Star>Fay Wray</Star>
6)          </Version>
7)          <Version year = "1976">
8)              <Star>Jeff Bridges</Star>
9)              <Star>Jessica Lange</Star>
10)         </Version>
11)         <Version year = "2005" />
12)     </Movie>
13)     <Movie title = "Footloose">
14)         <Version year = "1984">
15)             <Star>Kevin Bacon</Star>
16)             <Star>John Lithgow</Star>
17)             <Star>Sarah Jessica Parker</Star>
18)         </Version>
19)     </Movie>
20) </Movies>
```



-