

Advanced Databases Summative

Bradley Mackey

for 15th March 2019

1 Tree Structure

See the attached document for the tree diagram of `books.xml`.

2 Querying

2.1 XPath

2.1.1 Find all Mary Poppins books which were published before WWII.

1. **Query:**

```
//book[@publicationDate<"1939"] [title [contains(text(),"Mary Poppins")]]
```

2. **Interpretation:** This query first uses the Restricted Kleene Closure to find every occurrence of `<book>` in the XML structure. This ensures that we find all books at all levels of the database (if the database were to be grown in the future). We then use a tag predicate of `[@publicationDate<"1939"]` to filter out all books that were NOT released before 1939. For all the books returned by this, we then check if they have a `title` property. If they do, we use a nested predicate to check the title of the book. If the book title contains the string "Mary Poppins" (not including these quotation marks), the full book node will be returned.

3. **Limitations:** this query will only find Mary Poppins books if her name appears directly in the title. If such a Mary Poppins book existed that did not feature her name in the title, it would not be returned by our query. It is not sufficient just to check the author or illustrator as they may have been involved in creating other books (not Mary Poppins). Additionally, WWII started in September of 1939 and—as we have no way to distinguish the month of publication—it may be the case that a book published in 1939 was actually published before WWII. We have no way of telling this.

4. **Output:**

```
<book publicationDate="1934">
  <title> Mary Poppins </title>
  <author> P. L. Travers </author>
  <illustrator> Mary Shepard </illustrator>
</book>
```

```

<book publicationDate="1935">
  <title> Mary Poppins Comes Back </title>
  <author> P. L. Travers </author>
  <illustrator> Mary Shepard </illustrator>
</book>

```

2.1.2 Find all movies that have academy nominations.

1. Query:

```
//movie[normalize-space(academyNominations/text()) > "0"]
```

2. **Interpretation:** This query first uses the Restricted Kleene Closure to find every occurrence of <movie> in the XML structure. This ensures that we find all movies at all levels of the database (if the database were to be grown in the future). We then use a tag predicate of `academyNominations`, filtering out all movies that do not include an <academyNominations> node. We access the text stored at this node using the XPath `text()` function. As the XML files provided have whitespace surrounding the text in each node, we use the `normalize-space` function to remove this whitespace, as to not retrieve inaccurate results. We then ensure that the number of nominations is greater than 0, as it could be possible a movie could have an <academyNominations> node with 0 nominations. Movie nodes filtered out by this predicate are not included in our results.

3. Limitations:

4. Output:

```

<movie language="En" type="film" year="1964">
  <title> Mary Poppins </title>
  <actress> Julie Andrews </actress>
  <actor> Dick Van Dyke </actor>
  <actor> David Tomlison </actor>
  <actress> Glynis Johns </actress>
  <producer> Walt Disney </producer>
  <director> Robert Stevenson </director>
  <academyNominations> 13 </academyNominations>
</movie>
<movie language="En" type="film" year="2013">
  <title> Saving Mr. Banks </title>
  <actress> Emma Thompson </actress>
  <actor> Tom Hanks </actor>
  <academyNominations> 1 </academyNominations>
</movie>

```

2.2 XQuery

- 2.2.1 How many books have there been published about Mary Poppins?
- 2.2.2 List the books that have been published before the second Mary Poppins movie appeared.
- 2.2.3 How much later has the most recent Mary Poppins movie appeared, compared to the second book about Mary Poppins?