F28DM - Database Management Systems

Coursework 2: XML

Authors

Daniel Barker	-	H00250740
Tommy Lamb	-	H00217505

1 Division of work

The work for this coursework was split very evenly between both members. Each did two XPath expressions and one XSLT. More specifically, Mr Lamb contributed XPaths Party bookings and Gabbay Bookings, and Mr Barker Room Bookings and High Capacity Rooms. Mr Lamb developed the XML file itself with example data whilst Mr Barker developed the XML Schema. Both were naturally done in close association with one another.

2 XML Overview

The XML file created contains a selection of members' bookings with the associated room and equipment details slightly modified from our SQL coursework. A variety of bookings were created cover various bases, such as those with no equipment booked or more than one piece of equipment booked. The same is true of the rooms, where the cardinality of the relationship between rooms and equipment can be demonstrated.
Most of the booking data focuses on similar times and equipment to emulate a much larger file with significant amounts of bookings, though other dates and times are used to also demonstrate the scalability and spread of data the XML file can store.
In order to keep the data as versatile as possible, the data was normalised in the same manner as with a RDMBS. This means that the file has three predominant sub-trees: bookings, equipment, and rooms.

The bookings are contained within a date element, and each have their own unique booking ID (bID). Each has a start time, end time, associated room, and uses 0 or more pieces of equipment. A boolean also records if the booking is for an entire room.

The rooms are all stored together, identified by a room number (rnumber) attribute. Each has a defined maximum capacity, alongside pertinent details of any and all equipment contained within.

Each piece of equipment is identified by a code derived from its class and make or model (eID). The class, make, and model of each piece of equipment is recorded, alongside a small description field which could be presented to end-users of the booking system to inform their choices.

3 XPath Expressions

3.1 Party bookings

This XPath expression extracts the booking IDs (bID) of all bookings for more than 1 piece of equipment, of either the same or different make, model, and type.

```
//bookingdetail[count(./uses) > 1]/@bid | //uses[amount > 1]/../@bid
```

```
<!-- Results -->
<?xml version="1.0" encoding="UTF-8"?>
<result>
4
6
8
9
</result>
```

3.2 Gabbay Bookings

An expression which returns all bIDs for bookings on equipment of make ”Gabbay”. Gabbay is in this case simply a example, and could easily be replaced with any other name.

```
//uses[eid = //equipmentdetail[make = "Gabbay"]/@eid]/../@bid
```

```
<!-- Results -->
<?xml version="1.0" encoding="UTF-8"?>
<result>
4
8
</result>
```

3.3 Room Bookings

The following XPath returns all bIDs for room number 1, as an example. Such an expression can be useful when an employee wants to view bookings for a particular room, should it become suddenly unavailable or similar.

```
//bookingdetail[room = 1]/@bid
```

```
<!-- Results -->
<?xml version="1.0" encoding="UTF-8"?>
<result>
1
2
3
</result>
```

3.4 High Capacity Rooms

Returning all room numbers for rooms with a maximum capacity greater than or equal to 20, this expression can be particularly useful for making party bookings.

```
//roomdetail[maxcapacity>=20]/@rnumber
```

```
<!-- Results -->
<?xml version="1.0" encoding="UTF-8"?>
<result>
1
```

```
2
3
4
7
11
</result>
```