23.2

```
Emp = (ename, ChildrenSet setof(Children), SkillsSet setof(Skills))
Children = (name, Birthday)
Birthday = (day, month, year)
Skills = (type, ExamsSet setof(Exams))
Exams = (year, city)
```

```
<!DOCTYPE db [
<!ELEMENT emp (ename, children*, skills*)>
<!ELEMENT children (name, birthday)>
<!ELEMENT birthday (day, month, year)>
<!ELEMENT skills (type, exams+)>
<!ELEMENT exams (year, city)>
<!ELEMENT ename( #PCDATA )>
<!ELEMENT name( #PCDATA )>
<!ELEMENT day( #PCDATA )>
<!ELEMENT month( #PCDATA )>
<!ELEMENT year( #PCDATA )>
<!ELEMENT type( #PCDATA )>
<!ELEMENT city( #PCDATA )>
] >
```

23.3

/db/emp/skills/type

23.9

a.

```
<parts>
<part>
<name> bicycle </name>
<subpartinfo>
<part>
<name> wheel </name>
<subpartinfo>
<part>
```

```
<name> rim </name>
</part>
<qty> 1 </qty>
</subpartinfo>
<subpartinfo>
<part>
<name> spokes </name>
</part>
<qty> 40 </qty>
</subpartinfo>
<subpartinfo>
<part>
<name> tire </name>
</part>
<qty> 1 </qty>
</subpartinfo>
</part>
<qty> 2 </qty>
</subpartinfo>
<subpartinfo>
<part>
<name> brake </name>
</part>
<qty> 2 </qty>
</subpartinfo>
<subpartinfo>
<part>
<name> gear </name>
</part>
<qty> 3 </qty>
</subpartinfo>
<subpartinfo>
<part>
<name> frame </name>
</part>
<qty> 1 </qty>
</subpartinfo>
</part>
</parts>
```

b.

part(partid,name)
subpartinfo(partid, subpartid, qty)

Attributes partid and subpartid of subpartinfo are foreign keys to part.

c.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="parts" type="partsType" />
<xs:complexType name="partType">
<xs:sequence>
<xs:element name="name" type="xs:string"/>
<xs:element name="subpartinfo" type="subpartinfoType"
minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:complexType name="subpartinfoType"/>
<xs:sequence>
<xs:element name="part" type="partType"/>
<xs:element name="quantity" type="xs:string"/>
</xs:sequence>
</xs:schema>
```

23.11

a.

```
for $e in /db/emp,
$m in distinct-values($e/children/birthday/month)
where $m = 'March'
return $e/ename
```

b.

```
for $e in /db/emp
$s in $e/skills[type='typing']
$exam in $s/exams
where $exam/city= 'Dayton'
return $e/ename
```

c.

```
for $t in distinct-values (/db/emp/skills/type)
return $t
```