

Jawaban PR XML Database

13518149 Stefanus Gusega Gunawan

1. Berikut ini DTD yang dihasilkan untuk membentuk skema *database* pada soal.

```
<!DOCTYPE company [  
    <!ELEMENT company (department* , employee*, project*, emp_child*,  
emp_project*)>  
    <!ELEMENT department(Dname)>  
    <!ATTLIST department  
        DeptID ID #REQUIRED  
    >  
    <!ELEMENT employee(Name, Address_Street, Address_Num,  
Address_City, Address_Zipcode)>  
    <!ATTLIST employee  
        EmpID ID #REQUIRED  
    >  
    <!ELEMENT project(PName, Budget, StartDate, EndDate)>  
    <!ATTLIST project  
        ProjID ID #REQUIRED  
    >  
    <!ELEMENT emp_child(Birthday)>  
    <!ATTLIST emp_child  
        CName CDATA #REQUIRED  
        EmpID IDREF #REQUIRED  
    >  
    <!ELEMENT emp_project(Position)>  
    <!ATTLIST emp_project  
        ProjID IDREF #REQUIRED  
        EmpID IDREF #REQUIRED  
    >  
]
```

```

<!ELEMENT Dname(#PCDATA)>
<!ELEMENT Name(#PCDATA)>
<!ELEMENT Address_Street(#PCDATA)>
<!ELEMENT Address_Num(#PCDATA)>
<!ELEMENT Address_City(#PCDATA)>
<!ELEMENT Address_Zipcode(#PCDATA)>
<!ELEMENT Birthday(#PCDATA)>
<!ELEMENT PName(#PCDATA)>
<!ELEMENT Budget(#PCDATA)>
<!ELEMENT StartDate(#PCDATA)>
<!ELEMENT EndDate(#PCDATA)>
<!ELEMENT Position(#PCDATA)>
]>

```

Lalu, berikut ini XML Schema-nya.

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="company">
  <xs:complexType name="companyType">
    <xs:sequence>
      <xs:element ref="department" minOccurs="0"
maxOccurs="unbounded" />
      <xs:element ref="employee" minOccurs="0"
maxOccurs="unbounded" />
      <xs:element ref="emp_child" minOccurs="0"
maxOccurs="unbounded" />
      <xs:element ref="project" minOccurs="0"
maxOccurs="unbounded" />
      <xs:element ref="emp_project" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

</xs:complexType>

<xs:key name="deptKey">
    <xs:selector xpath = "/company/department"/>
    <xs:field xpath = "DeptID"/>
</xs:key>

<xs:key name="empKey">
    <xs:selector xpath = "/company/employee"/>
    <xs:field xpath = "EmpID"/>
</xs:key>

<xs:key name="projKey">
    <xs:selector xpath = "/company/project"/>
    <xs:field xpath = "ProjID"/>
</xs:key>

<xs:keyref name="childEmpFKey" refer="empKey">
    <xs:selector xpath = "/company/emp_child"/>
    <xs:field xpath = "EmpID"/>
</xs:keyref>

<xs:keyref name="projectEmpFKey" refer="empKey">
    <xs:selector xpath = "/company/emp_project"/>
    <xs:field xpath = "EmpID"/>
</xs:keyref>

<xs:keyref name="empProjFKey" refer="projKey">
    <xs:selector xpath = "/company/emp_project"/>
    <xs:field xpath = "ProjID"/>
</xs:keyref>
</xs:element>

<xs:element name="department">
    <xs:complexType>

```

```
<xs:sequence>
    <xs:element name="Dname" type="xs:string" />
</xs:sequence>
    <xs:attribute name="DeptID" type="xs:ID" use="required"/>
</xs:complexType>
</xs:element>

<xs:element name="employee">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Name" type="xs:string" />
            <xs:element name="Address_Street" type="xs:string" />
            <xs:element name="Address_Num" type="xs:string" />
            <xs:element name="Address_City" type="xs:string" />
            <xs:element name="Address_Zipcode" type="xs:string" />
        </xs:sequence>
        <xs:attribute name="EmpID" type="xs:ID" use="required"/>
    </xs:complexType>
</xs:element>

<xs:element name="emp_child">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Birthday" type="xs:string" />
        </xs:sequence>
        <xs:attribute name="EmpID" type="xs:IDREF" use="required"/>
        <xs:attribute name="CName" type="xs:string" use="required"/>
    </xs:complexType>
</xs:element>
```

```

<xs:element name="project">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PName" type="xs:string" />
      <xs:element name="Budget" type="xs:decimal" />
      <xs:element name="StartDate" type="xs:date" />
      <xs:element name="EndDate" type="xs:date" />
    </xs:sequence>
    <xs:attribute name="ProjID" type="xs:ID" use="required"/>
  </xs:complexType>
</xs:element>

<xs:element name="emp_project">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Position" type="xs:string" />
    </xs:sequence>
    <xs:attribute name="EmpID" type="xs:IDREF" use="required"/>
    <xs:attribute name="ProjID" type="xs:IDREF" use="required"/>
  </xs:complexType>
</xs:element>

```

2. Lalu berikut ini adalah *query-query*.

- a. XPath, list all children of employee with ID="E1234"

```
/company/emp_child[@EmpID = "E1234"]
```

- b. XQuery, list all projects and the names of employees assigned to the project

```

for $p in /company/project
return
<project ProjID="{ $p/@ProjID}">
  { $p/* }
  { for $ep in /company/emp_project[@ProjID = $p/@ProjID],

```

```
    $e in /company/employee[@EmpID = $ep/@EmpID]
  return <employee_name> {$e/Name/text()} </employee_name>}
</project>
```

- c. XQuery, list all departments and the number of employees for each department.
Asumsi: Employee memiliki foreign key ke department bernama DeptID.

```
for $d in /company/department
return
<department DeptID="{ $d/@DeptID}">
  { $d/* }
  <total_emp>
    {
      count(/company/employee[@DeptID = $d/@DeptID])
    }
  </total_emp>
</department>
```