

JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Készítette: **Szilágyi Gábor**

Neptunkód: **S9NJK6**

A féléves beadandó feladat egy XML adatbázis megtervezése, létrehozása és ennek az adminisztrálásához szükséges kód létrehozása.

A saját témámnak egy egyszerű kezdetleges RPG játéknak az adatbázisát választottam. Ennek főbb részei a felhasználói fiók, ranglista, karakterek statisztikája és egy baráti lista. Ezek az egyedek főként a tőlük elvárt és egyértelmű tulajdonságokkal bírnak. Ezek közé tartozik pl.: az e-mail cím, karakterek neve, elérhetőségi státusz, rangbeli helyezés stb.

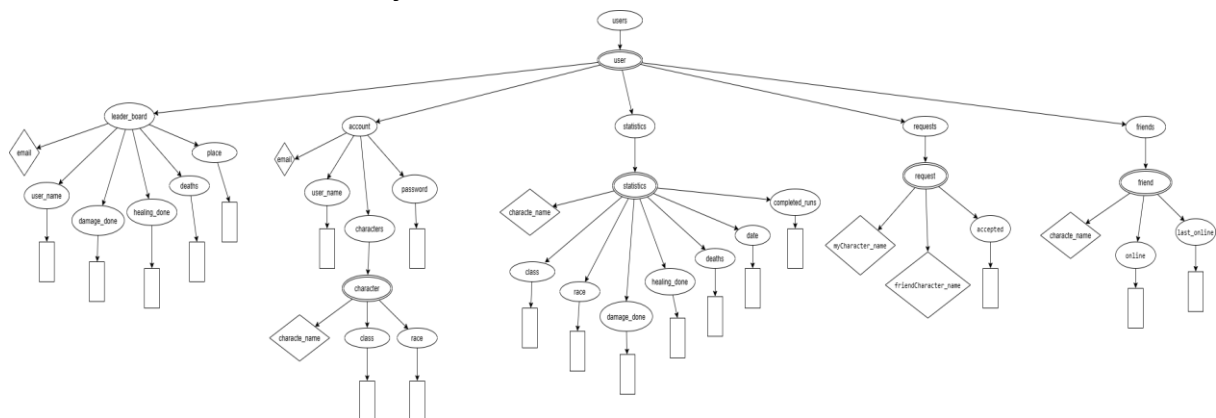
A relációs modellhez képest a megvalósításnál kisebb változásokat voltam képtelen alkalmazni a könnyeb átláthatóság, megértés és az adatbás megkötései végett.

Az adatbázist adminisztráló programok az alábbi három féle műveletet foglalják magukba: adatok beolvasása, adatok módosítása, adatok lekérdezése.

Az adatbázis ER modellje:



Az adatbázis XDM modellje:



Az XDM modell annyiban különbözik az ER modelltől, hogy egy egybefoglaló users névteret kellett bevezetni amin belül találhatóak meg a user példányok. A useren belül találhatóak meg az ER modellben már látott egyedek és tulajdonságaik a request kapcsolótáblával együtt.

A rombuszokban lehet látni a különböző kulcsokat amiknek főként két fajtája van, a fiókhoz tartozó e-mail cím és a karaktereknek a nevei. A karakter neveken belül két félért különböztetünk meg az egyik a saját karaktereink neve a másik pedig a karakterünk barátainak a karakter neve.

Az XML dokumentum kódja:

```
<?xml version="1.0" encoding="UTF-8"?>
<users xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="XMLSchemaS9NJK6.xsd">
  <user>
    <account email="sajt2@sajt.sa">
      <user_name>Sajtoska2</user_name>
      <password>sasajt</password>
      <characters>
        <character character_name="Gátierő">
          <class>Warrior</class>
          <race>Human</race>
        </character>
        <character character_name="Legoolass">
          <class>Archer</class>
          <race>Elf</race>
        </character>
      </characters>
    </account>
    <statistics>
      <statistic character_name="Gátierő">
        <class>Warrior</class>
        <race>Human</race>
        <damage_done>10000</damage_done>
        <healing_done>100</healing_done>
        <deaths>5</deaths>
        <completed_runs>10</completed_runs>
        <date>2021-11-02</date>
      </statistic>
      <statistic character_name="Legoolass">
        <class>Archer</class>
        <race>Elf</race>
        <damage_done>15000</damage_done>
        <healing_done>400</healing_done>
        <deaths>3</deaths>
        <completed_runs>5</completed_runs>
        <date>2021-11-10</date>
      </statistic>
      <statistic character_name="Legoolass">
        <class>Archer</class>
        <race>Elf</race>
        <damage_done>15000</damage_done>
```

```

        <healing_done>400</healing_done>
        <deaths>3</deaths>
        <completed_runs>5</completed_runs>
        <date>2021-11-09</date>
    </statistic>

</statistics>
<leader_board email="sajt2@sajt.sa">
    <user_name>Sajtoska2</user_name>
    <damage_done>25000</damage_done>
    <healing_done>500</healing_done>
    <deaths>8</deaths>
    <place>1</place>
</leader_board>
<requests>
    <request myCharacter_name="Legoolass"
friendCharacter_name="Maho">
        <accepted>true</accepted>
    </request>
    <request myCharacter_name="Legoolass"
friendCharacter_name="Wall">
        <accepted>false</accepted>
    </request>
</requests>
<friends>
    <friend character_name="Maho">
        <online>false</online>
        <last_online>2021-11-12</last_online>
    </friend>
</friends>
</user>
<user>
    <account email="asd@asd.as">
        <user_name>AsdDsa</user_name>
        <password>pontosanaz</password>
        <characters>
            <character character_name="Wall">
                <class>Tank</class>
                <race>Dwarf</race>
            </character>
        </characters>
    </account>
    <statistics>
        <statistic character_name="Wall">
            <class>Tank</class>
            <race>Dwarf</race>
            <damage_done>5000</damage_done>
            <healing_done>5000</healing_done>
            <deaths>2</deaths>
            <completed_runs>5</completed_runs>
            <date>2021-11-11</date>
        </statistic>
    </statistics>
    <leader_board email="asd@asd.as">
        <user_name>AsdDsa</user_name>
        <damage_done>5000</damage_done>
        <healing_done>5000</healing_done>
        <deaths>2</deaths>
        <place>2</place>
    </leader_board>
</requests>

```

```

        </requests>
        <friends>

        </friends>
    </user>
</users>

```

Az XML Schema kódja:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="qualified">
<xs:element name="users">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="user" type="userType" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <!--Keys-->
    <xs:key name="accountEmail">
        <xs:selector xpath="./user/account"/>
        <xs:field xpath="@email"/>
    </xs:key>

    <xs:key name="myCharacterName">
        <xs:selector xpath="./user/account/characters/character"/>
        <xs:field xpath="@character_name"/>
    </xs:key>

    <xs:key name="friendsCharacterName">
        <xs:selector xpath="./user/friends/friend"/>
        <xs:field xpath="@character_name"/>
    </xs:key>

    <xs:keyref name="statisticsRef" refer="myCharacterName">
        <xs:selector xpath="./user/statistics/statistic"/>
        <xs:field xpath="@character_name"/>
    </xs:keyref>

    <xs:keyref name="requestMyCharacterName" refer="myCharacterName">
        <xs:selector xpath="./user/requests/request"/>
        <xs:field xpath="@myCharacterName"/>
    </xs:keyref>

    <xs:keyref name="requestFriendCharacterName"
refer="friendsCharacterName">
        <xs:selector xpath="./user/requests/request"/>
        <xs:field xpath="@friendsCharacterName"/>
    </xs:keyref>

    <xs:keyref name="leaderBoardEmail" refer="accountEmail">
        <xs:selector xpath="./user/leader_board"/>
        <xs:field xpath="@email"/>
    </xs:keyref>

</xs:element>

```

```

<xs:complexType name="userType">
  <xs:sequence>
    <xs:element name="account" type="accountType"/>
    <xs:element name="statistics" type="statisticsType"
maxOccurs="unbounded"/>
    <xs:element name="leader_board" type="leader_boardType"/>
    <xs:element name="requests" type="requestsType"/>
    <xs:element name="friends" type="friendsType"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!--Simple myTypes-->
<xs:simpleType name="emailType">
  <xs:restriction base="xs:string">
    <xs:pattern value="^[^@]+@[^.]+\.[^.+\s]"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="classesType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Warrior"/>
    <xs:enumeration value="Archer"/>
    <xs:enumeration value="Mage"/>
    <xs:enumeration value="Tank"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="racesType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Human"/>
    <xs:enumeration value="Elf"/>
    <xs:enumeration value="Dwarf"/>
    <xs:enumeration value="Demon"/>
  </xs:restriction>
</xs:simpleType>

<!--Account elements-->
<xs:complexType name="accountType">
  <xs:sequence>
    <xs:element name="user_name" type="xs:string"/>
    <xs:element name="password" type="xs:string"/>
    <xs:element name="characters" type="charactersType"/>
  </xs:sequence>
  <xs:attribute name="email" type="emailType" use="required"/>
</xs:complexType>

<xs:complexType name="charactersType">
  <xs:sequence>
    <xs:element name="character" type="characterType"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="characterType">
  <xs:sequence>
    <xs:element name="class" type="classesType"/>
    <xs:element name="race" type="racesType"/>
  </xs:sequence>
  <xs:attribute name="character_name" type="xs:string"

```

```

use="required"/>
</xs:complexType>

<!--Statistics elements-->
<xs:complexType name="statisticsType">
  <xs:sequence>
    <xs:element name="statistic" type="statisticType"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="statisticType">
  <xs:sequence>
    <xs:element name="class" type="classesType"/>
    <xs:element name="race" type="racesType"/>
    <xs:element name="damage_done" type="xs:int"/>
    <xs:element name="healing_done" type="xs:int"/>
    <xs:element name="deaths" type="xs:int"/>
    <xs:element name="completed_runs" type="xs:int"/>
    <xs:element name="date" type="xs:date"/>
  </xs:sequence>
  <xs:attribute name="character_name" type="xs:string"
use="required"/>
</xs:complexType>
<!-- Leader board elements-->
<xs:complexType name="leader_boardType">
  <xs:sequence>
    <xs:element name="user_name" type="xs:string" />
    <xs:element name="damage_done" type="xs:int"/>
    <xs:element name="healing_done" type="xs:int"/>
    <xs:element name="deaths" type="xs:int"/>
    <xs:element name="place" type="xs:int"/>
  </xs:sequence>
  <xs:attribute name="email" type="emailType" use="required"/>
</xs:complexType>

<!--Request elements, connects Account with Friends-->
<xs:complexType name="requestsType">
  <xs:sequence>
    <xs:element name="request" type="requestType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="requestType">
  <xs:sequence>
    <xs:element name="accepted" type="xs:boolean"/>
  </xs:sequence>
  <xs:attribute name="myCharacter_name" type="xs:string"
use="required"/>
  <xs:attribute name="friendCharacter_name" type="xs:string"
use="required"/>
</xs:complexType>

<!--Friends elements (character_name=friend's character_name)-->
<xs:complexType name="friendsType">
  <xs:sequence>
    <xs:element name="friend" type="friendType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>

```



```

"+userNode.getNodeName()+"\n");

        //Account
        NodeList
accountNodeList=userElement.getElementsByTagName("account");
        Node accountNode=accountNodeList.item(0);
        Element accountElement=(Element)accountNode;

        String
accountEmail=accountElement.getAttribute("email");
        String
accountUserName=accountElement.getElementsByTagName("user_name").item(0).get
getTextContent();
        String
password=accountElement.getElementsByTagName("password").item(0).getTextCon
tent();

        System.out.println("Current Element:
"+accountNode.getNodeName());
        System.out.println("E-mail address: "+accountEmail);
        System.out.println("User name: "+accountUserName);
        System.out.println("Password: "+password+"\n");

        //Characters
        NodeList
charactersNodeList=accountElement.getElementsByTagName("character");
        for (int j=0;j<charactersNodeList.getLength();j++){
            Node characterNode=charactersNodeList.item(j);
            Element characterElement=(Element) characterNode;

            String
characterName=characterElement.getAttribute("character_name");
            String
characterClass=characterElement.getElementsByTagName("class").item(0).getTe
xtContent();
            String
race=characterElement.getElementsByTagName("race").item(0).getTextContent()
;

            System.out.println("Current Element:
"+characterNode.getNodeName());
            System.out.println("Name: "+characterName);
            System.out.println("Class: "+characterClass);
            System.out.println("Race: "+race+"\n");

        }

        //Statistics
        NodeList
statisticsNodeList=userElement.getElementsByTagName("statistic");

        for (int j=0;j<statisticsNodeList.getLength();j++){
            Node statisticNode=statisticsNodeList.item(j);
            Element statisticElement=(Element) statisticNode;

            String
characterName=statisticElement.getAttribute("character_name");
            String
characterClass=statisticElement.getElementsByTagName("class").item(0).getTe
xtContent();

```

```

        String
race=statisticElement.getElementsByTagName("race").item(0).getTextContent()
;

        String
damage=statisticElement.getElementsByTagName("damage_done").item(0).getTextContent();

        String
healing=statisticElement.getElementsByTagName("healing_done").item(0).getTextContent();

        String
deaths=statisticElement.getElementsByTagName("deaths").item(0).getTextContent();

        String
runs=statisticElement.getElementsByTagName("completed_runs").item(0).getTextContent();

        String
date=statisticElement.getElementsByTagName("date").item(0).getTextContent();
;

        System.out.println("Current Element:
"+statisticNode.getNodeName());
        System.out.println("Name: "+characterName);
        System.out.println("Class: "+characterClass);
        System.out.println("Race: "+race);
        System.out.println("Damage done: "+damage);
        System.out.println("Healing done: "+healing);
        System.out.println("Deaths: "+deaths);
        System.out.println("Completed runs: "+runs);
        System.out.println("Date: "+date+"\n");
    }

    //Leader board
    NodeList
leaderBoardNodeList=userElement.getElementsByTagName("leader_board");
    Node leaderBoardNode=leaderBoardNodeList.item(0);
    Element leaderBoardElement=(Element) leaderBoardNode;

        String
leaderEmail=leaderBoardElement.getAttribute("email");
        String
leaderUserName=leaderBoardElement.getElementsByTagName("user_name").item(0).getTextContent();
        String
leaderDamage=leaderBoardElement.getElementsByTagName("damage_done").item(0).getTextContent();
        String
leaderHealing=leaderBoardElement.getElementsByTagName("healing_done").item(0).getTextContent();
        String
leaderDeaths=leaderBoardElement.getElementsByTagName("deaths").item(0).getTextContent();
        String
place=leaderBoardElement.getElementsByTagName("place").item(0).getTextContent();

        System.out.println("Current Element:
"+leaderBoardNode.getNodeName());
        System.out.println("E-mail address: "+leaderEmail);
        System.out.println("User name: "+leaderUserName);
        System.out.println("Damage done: "+leaderDamage);
        System.out.println("Healing done: "+leaderHealing);

```

```

        System.out.println("Deaths: "+leaderDeaths);
        System.out.println("Place: "+place+"\n");

        //Requests
        NodeList
requestsNodeList=userElement.getElementsByTagName("request");
        for (int j=0;j< requestsNodeList.getLength();j++){
            Node requestNode=requestsNodeList.item(j);
            Element requestElement=(Element) requestNode;

            String
myCharacterName=requestElement.getAttribute("myCharacter_name");
            String
friendCharacterName=requestElement.getAttribute("friendCharacter_name");
            String
accepted=requestElement.getElementsByTagName("accepted").item(0).getTextCon
tent();

            System.out.println("Current Element:
"+requestNode.getNodeName());
            System.out.println("My character name:
"+myCharacterName);
            System.out.println("Friend character name:
"+friendCharacterName);
            System.out.println("Accepted: "+accepted+"\n");
        }

        //Friends
        NodeList
friendsNodeList=userElement.getElementsByTagName("friend");
        for (int j=0;j< friendsNodeList.getLength();j++){
            Node friendNode=friendsNodeList.item(j);
            Element friendElement=(Element) friendNode;

            String
characterName=friendElement.getAttribute("character_name");
            String
online=friendElement.getElementsByTagName("online").item(0).getTextContent(
);
            String
lastOnline=friendElement.getElementsByTagName("last_online").item(0).getTex
tContent();

            System.out.println("Current Element:
"+friendNode.getNodeName());
            System.out.println("Character name:
"+characterName);
            System.out.println("Online: "+online);
            System.out.println("Last online:
"+lastOnline+"\n");
        }

    }

}

} catch (SAXException | IOException | ParserConfigurationException e)
{
    System.err.println(e.getMessage());
}

```

```

    } catch (NumberFormatException nfe) {
        System.err.println(nfe.getMessage());
    }
}
}

```

Adatmódosítás (DomModifyS9NJK6.java):

```

package hu.domparsing.S9NJK6;

import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.*;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import javax.xml.xpath.XPath;
import javax.xml.xpath.XPathConstants;
import javax.xml.xpath.XPathExpressionException;
import javax.xml.xpath.XPathFactory;
import java.io.IOException;

public class DomModifyS9NJK6 {
    public static void main(String[] args) {
        Document doc;
        DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();

        try {
            DocumentBuilder db = dbf.newDocumentBuilder();
            doc = db.parse("src/hu/domparsing/S9NJK6/XMLS9NJK6.xml");
            doc.getDocumentElement().normalize();
            XPath xPath = XPathFactory.newInstance().newXPath();
            Node rootNode = doc.getChildNodes().item(0);

            //Character Gátierő's name change to MuscleFbrain
            NodeList characterNamesNodeList=(NodeList)
xPath.compile("//*[@character_name]").evaluate(doc,
XPathConstants.NODESET);
            NodeList friendCharNameNodeList=(NodeList)
xPath.compile("//*[@friendCharacter_name]").evaluate(doc,
XPathConstants.NODESET);
            NodeList myCharNameNodeList=(NodeList)
xPath.compile("//*[@myCharacter_name]").evaluate(doc,
XPathConstants.NODESET);

            changeAttribute(characterNamesNodeList,"character_name","Gátierő","MuscleFb
rain");

            changeAttribute(friendCharNameNodeList,"friendCharacter_name","Gátierő","Mu
scleFbrain");

```

```
changeAttribute(myCharNameNodeList,"myCharacter_name","Gátieró","MuscleFbra  
in");
```

```
//Change character Legoolass's damage done from 15000 to  
16000,rund from 5 to 6 and deaths from 3 to 4 from in statistics with the  
date 2021-11-10
```

```
NodeList damageChangeNodeList=(NodeList)  
XPath.compile("//statistic[@character_name='Legoolass']").evaluate(doc,  
XPathConstants.NODESET);  
for (int i = 0; i < damageChangeNodeList.getLength(); i++) {  
  
    Node characterNode = damageChangeNodeList.item(i);  
  
    if(characterNode.getNodeType() == Node.ELEMENT_NODE) {  
        Element characterElement = (Element) characterNode;  
  
        if  
(characterElement.getElementsByTagName("date").item(0).getTextContent().equ  
als("2021-11-10")) {  
            NodeList  
damageNodeList=characterElement.getElementsByTagName("damage_done");  
  
            for (int j=0;j< damageNodeList.getLength();j++){  
                Node damageNode=damageNodeList.item(j);  
  
                if  
(damageNode.getNodeType()==Node.ELEMENT_NODE){  
                    damageNode.setTextContent("16000");  
  
                }  
  
            }  
            NodeList  
deathsNodeList=characterElement.getElementsByTagName("deaths");  
  
            for (int j=0;j< deathsNodeList.getLength();j++){  
                Node deathsNode=deathsNodeList.item(j);  
  
                if  
(deathsNode.getNodeType()==Node.ELEMENT_NODE){  
                    deathsNode.setTextContent("4");  
  
                }  
  
            }  
            NodeList  
runsNodeList=characterElement.getElementsByTagName("completed_runs");  
  
            for (int j=0;j< runsNodeList.getLength();j++){  
                Node runsNode=runsNodeList.item(j);  
  
                if (runsNode.getNodeType()==Node.ELEMENT_NODE){  
                    runsNode.setTextContent("6");  
  
                }  
  
            }  
  
        }  
    }  
}
```

```

        }
    }
    Transformer tf =
TransformerFactory.newInstance().newTransformer();

    tf.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
    tf.setOutputProperty(OutputKeys.INDENT, "yes");
    tf.setOutputProperty("{http://xml.apache.org/xslt}indent-
amount", "2");

    DOMSource source = new DOMSource(rootNode);
    StreamResult console = new StreamResult(System.out);

    System.out.println("----- Modified File -----
-----");
    tf.transform(source, console);

    }catch(SAXException | IOException | ParserConfigurationException |
NumberFormatException e) {
        System.err.println(e.getMessage());

    } catch (XPathExpressionException e) {
        e.printStackTrace();
    } catch (TransformerException e) {
        e.printStackTrace();
    }
}

    public static void changeAttribute(NodeList nodeList,String
attributeName,String from,String to){
        for (int i=0;i< nodeList.getLength();i++){

            Node charNamesNode=nodeList.item(i);

            if(charNamesNode.getNodeType() == Node.ELEMENT_NODE) {
                Element charNameElement=(Element) charNamesNode;
                if
(charNameElement.getAttribute(attributeName).equals(from)){
                    charNameElement.setAttribute(attributeName,to);
                }
            }
        }
    }
}

```

Adatlekérdezés (DomQueryS9NJK6.java):

```

package hu.domparsed.S9NJK6;

import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;

```

```

import javax.xml.xpath.XPath;
import javax.xml.xpath.XPathConstants;
import javax.xml.xpath.XPathExpressionException;
import javax.xml.xpath.XPathFactory;
import java.io.IOException;

public class DomQueryS9NJK6 {
    public static void main(String[] args) {
        Document doc;
        DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();

        try {
            DocumentBuilder db = dbf.newDocumentBuilder();
            doc = db.parse("src/hu/domparse/S9NJK6/XMLS9NJK6.xml");
            doc.getDocumentElement().normalize();

            System.out.println("Root element:
"+doc.getDocumentElement().getNodeName()+"\n-----\n");

            XPath xPath = XPathFactory.newInstance().newXPath();

            String accountInfos="//account";

            //Accounts e-mail, user name, password
            NodeList accountNodeList=(NodeList)
xPath.compile(accountInfos).evaluate(doc,XPathConstants.NODESET);

            for (int i=0;i< accountNodeList.getLength();i++) {
                Node accountNode = accountNodeList.item(i);

                if (accountNode.getNodeType() == Node.ELEMENT_NODE &&
accountNode.getNodeName().equals("account")) {
                    Element accountElement=(Element) accountNode;

                    String
accountEmail=accountElement.getAttribute("email");
                    String
accountUserName=accountElement.getElementsByTagName("user_name").item(0).ge
tTextContent();
                    String
password=accountElement.getElementsByTagName("password").item(0).getTextCon
tent();

                    System.out.println("Current Element:
"+accountNode.getNodeName());
                    System.out.println("E-mail address: "+accountEmail);
                    System.out.println("User name: "+accountUserName);
                    System.out.println("Password: "+password+"\n");
                }

            }

            //Greather or equal than 12000 damage done in statistic with
character name
            NodeList damageNodeList= (NodeList)
xPath.compile("//statistic/damage_done").evaluate(doc,XPathConstants.NODESE
T);

            for (int i=0;i< damageNodeList.getLength();i++) {

```

```

        Node damageDone = damageNodeList.item(i);
        int damage = Integer.parseInt(damageDone.getTextContent());
        Element damageElement = (Element) damageDone;
        if (damage >= 12000){
            Element nameElement = (Element)
damageElement.getParentNode();
            System.out.println("Character: " +
nameElement.getAttribute("character_name"));
            System.out.println("Damage done: " + damage + "\n");
        }
    }

    } catch (SAXException | IOException | ParserConfigurationException |
NumberFormatException e) {
        System.err.println(e.getMessage());

    } catch (XPathExpressionException e) {
        e.printStackTrace();
    }
}
}

```


Outputok:

DomReadS9NJK6.java:

```
Root Element:users
-----
Current Element: user

Current Element: account
E-mail address: saj2@sajt.sa
User name: Sajtoska2
Password: sasajt

Current Element: character
Name: Gátierő
Class: Warrior
Race: Human

Current Element: character
Name: Legoolass
Class: Archer
Race: Elf

Current Element: statistic
Name: Gátierő
Class: Warrior
Race: Human
Damage done: 10000
Healing done: 100
Deaths: 5
Completed runs: 10
Date: 2021-11-02

Current Element: statistic
Name: Legoolass
Class: Archer
Race: Elf
Damage done: 15000
Healing done: 400
Deaths: 3
Completed runs: 5
Date: 2021-11-10
```

Current Element: statistic

Name: Legoolass

Class: Archer

Race: Elf

Damage done: 15000

Healing done: 400

Deaths: 3

Completed runs: 5

Date: 2021-11-09

Current Element: friend

Character name: Maho

Online: false

Last online: 2021-11-12

Current Element: user

Current Element: leader_board

E-mail address: saj2@sajt.s

User name: Sajtoska2

Damage done: 25000

Healing done: 500

Deaths: 8

Place: 1

Current Element: account

E-mail address: asd@asd.as

User name: AsdDsa

Password: pontosanaz

Current Element: character

Name: Wall

Class: Tank

Race: Dwarf

Current Element: request

My character name: Legoolass

Friend character name: Maho

Accepted: true

Current Element: statistic

Name: Wall

Class: Tank

Race: Dwarf

Current Element: request

My character name: Legoolass

Friend character name: Wall

Accepted: false

Damage done: 5000

Healing done: 5000

Deaths: 2

Completed runs: 5

Date: 2021-11-11

Current Element: friend

Character name: Maho

Online: false

Last online: 2021-11-12

Current Element: leader_board

E-mail address: asd@asd.as

User name: AsdDsa

Damage done: 5000

Healing done: 5000

Deaths: 2

Place: 2

Current Element: account

E-mail address: asd@asd.as

User name: AsdDsa

Password: pontosanaz

DomModifyS9NJK6.java:

```
"C:\Program Files\Java\jdk-14.0.1\bin\java.exe" --javaagent:C:\Users\baosz\AppData\Local\JetBrains\IntelliJ IDEA 2021.2.2\lib\idea_rt.jar=50092:C:\Users\
----- Modified File -----
<?xml version="1.0" encoding="UTF-8"?><users xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="XMLSchemaS9NJK6.xsd">

  <user>

    <account email="sajt2@sajt.sa">

      <user_name>Sajtoska2</user_name>

      <password>sasajt</password>

      <characters>

        <character character_name="MuscleFbrain">

          <class>Warrior</class>

          <race>Human</race>

        </character>

        <character character_name="Legoolass">

          <class>Archer</class>

          <race>Elf</race>

        </character>

      </characters>

    </account>

    <statistics>

      <statistic character_name="MuscleFbrain">

        <class>Warrior</class>|
```

```
<race>Human</race>
```

```
<damage_done>10000</damage_done>
```

```
<healing_done>100</healing_done>
```

```
<deaths>5</deaths>
```

```
<completed_runs>10</completed_runs>
```

```
<date>2021-11-02</date>
```

```
</statistic>
```

```
<statistic character_name="Legoolass">
```

```
<class>Archer</class>
```

```
<race>Elf</race>
```

```
<damage_done>16000</damage_done>
```

```
<healing_done>400</healing_done>
```

```
<deaths>4</deaths>
```

```
<completed_runs>6</completed_runs>
```

```
<date>2021-11-10</date>
```

```
</statistic>
```

```
<statistic character_name="Legoolass">
```

```
<class>Archer</class>
```

```
<race>Elf</race>
```

```
<damage_done>15000</damage_done>
```

```
<damage_done>15000</damage_done>

<healing_done>400</healing_done>

<deaths>3</deaths>

<completed_runs>5</completed_runs>

<date>2021-11-09</date>

</statistic>

</statistics>

<leader_board email="sajt2@sajt.sa">

  <user_name>Sajtoska2</user_name>

  <damage_done>25000</damage_done>

  <healing_done>500</healing_done>

  <deaths>8</deaths>

  <place>1</place>

</leader_board>

<requests>

  <request friendCharacter_name="Maho" myCharacter_name="Legoolass">

    <accepted>true</accepted>

  </request>

  <request friendCharacter_name="Wall" myCharacter_name="Legoolass">

    <accepted>false</accepted>
```

```
Dommodirys9910KO x
<accepted>false</accepted>

</request>

</requests>

<friends>

  <friend character_name="Maho">

    <online>false</online>

    <last_online>2021-11-12</last_online>

  </friend>

</friends>

</user>

<user>

  <account email="asd@asd.as">

    <user_name>AsdDsa</user_name>

    <password>pontosanaz</password>

    <characters>

      <character character_name="Wall">

        <class>Tank</class>

        <race>Dwarf</race>

      </character>

    </characters>

  </account>
```

</account>

<statistics>

<statistic character_name="Wall">

<class>Tank</class>

<race>Dwarf</race>

<damage_done>5000</damage_done>

<healing_done>5000</healing_done>

<deaths>2</deaths>

<completed_runs>5</completed_runs>

<date>2021-11-11</date>

</statistic>

</statistics>

<leader_board email="asd@asd.as">

<user_name>AsdDsa</user_name>

<damage_done>5000</damage_done>

<healing_done>5000</healing_done>

<deaths>2</deaths>

<place>2</place>

</leader_board>

<requests>

```
<healing_done>5000</healing_done>

    <deaths>2</deaths>

    <completed_runs>5</completed_runs>

    <date>2021-11-11</date>

</statistic>

</statistics>

<leader_board email="asd@asd.as">

    <user_name>AsdDsa</user_name>

    <damage_done>5000</damage_done>

    <healing_done>5000</healing_done>

    <deaths>2</deaths>

    <place>2</place>

</leader_board>

<requests>

    </requests>

<friends>

    </friends>

</user>

</users>
```

Process finished with exit code 0

DomQueryS9NJK6.java:

```
Root element: users
```

```
-----
```

```
Current Element: account
```

```
E-mail address: saj2@sajt.sa
```

```
User name: Sajtoska2
```

```
Password: sasajt
```

```
Current Element: account
```

```
E-mail address: asd@asd.as
```

```
User name: AsdDsa
```

```
Password: pontosanaz
```

```
Character: Legoolass
```

```
Damage done: 15000
```

```
Character: Legoolass
```

```
Damage done: 15000
```