

Adatkezelés XML-ben

2. Beadandó

Készítette: Fehér Máté János
Neptun kód: BZ33U3

```

package uni.sajat;

import org.w3c.dom.*;
import org.xml.sax.SAXException;

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

import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerException;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;

import java.io.File;
import java.io.IOException;
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        fajlokListazasa();
        System.out.println("Írjon be egy .xml kiterjesztésű fájlt, kiterjesztés nélkül!");
        String fajl = sc.next();
        File inputFile = new File(fajl+".xml");
        DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = null;
        try {
            dBuilder = dbFactory.newDocumentBuilder();
        } catch (ParserConfigurationException e) {
            e.printStackTrace();
        }
        Document doc = null;
        try {
            doc = dBuilder.parse(inputFile);
        } catch (SAXException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        doc.getDocumentElement().normalize();
        System.out.println("Root element: " + doc.getDocumentElement().getNodeName());
        mainloop:
        while(true) {
            System.out.println("Válassza ki a menüpontot (1-4): ");
            System.out.println("1) Listázás");
            System.out.println("2) Feltöltés");
            System.out.println("3) Módosítás");
            System.out.println("4) Kilépés");

            int input = sc.nextInt();
            int input2 = 0;

```

```

        switch (input) {
            case 1:
                System.out.println("Mit akar listázni? (1-4)");
                System.out.println("1) Teljes dokumentum");
                System.out.println("2) Vezetők listája");
                System.out.println("3) Részlegek listája");
                System.out.println("4) Alkalmazottak listája");
                System.out.println("5) Vissza...");
                input2 = sc.nextInt();
                switch (input2) {
                    case 1:
                        dokumentumListazasa(doc);
                        break;
                    case 2:
                        vezetokListazasa(doc);
                        break;
                    case 3:
                        reszlegekListazasa(doc);
                        break;
                    case 4:
                        alkalmazottakListazasa(doc);
                        break;
                    case 5:
                        break;
                    default:
                        System.out.println("Nincs ilyen opció.");
                }
                break;
            case 2:
                try {
                    feltoltes(doc);
                } catch (TransformerException e) {
                    e.printStackTrace();
                }
                break;
            case 3:
                try {
                    modosit(doc);
                } catch (TransformerException e) {
                    e.printStackTrace();
                }
                break;
            case 4:
                break mainloop;
            default:
                System.out.println("Nincs ilyen opció.");
        }

    }

}

private static void fajlokListazasa() {
    File folder = new File(".");
    File[] listOfFiles = folder.listFiles();
}

```

```

        System.out.println("Elérhető fájlok: ");
        for (File file : listOfFiles) {
            if (file.isFile()) {
                System.out.println("\t" + file.getName());
            }
        }
    }

    private static void dokumentumListazasa(Document doc) {
        vezetokListazasa(doc);
        reszlegekListazasa(doc);
        alkalmazottakListazasa(doc);
    }

    private static void vezetokListazasa(Document doc) {
        NodeList nList = doc.getElementsByTagName("Vezeto");
        System.out.println("-----");

        for (int temp = 0; temp < nList.getLength(); temp++) {
            Node nNode = nList.item(temp);
            System.out.println("\nCurrent Element : " +
nNode.getNodeName());

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element eElement = (Element) nNode;
                System.out.println("VezetoID : "
                    + eElement.getAttribute("VezetoID"));
                System.out.println("E-Mail : "
                    + eElement
                        .getElementsByTagName("E-Mail")
                        .item(0)
                        .getTextContent());
                System.out.println("Vezeteknev : "
                    + eElement
                        .getElementsByTagName("Vezeteknev")
                        .item(0)
                        .getTextContent());

                for(int
i=0;i<eElement.getElementsByTagName("Keresztnev").getLength();i++){
                    System.out.println("Keresztnev" + (i+1) + " : "
                        + eElement
                            .getElementsByTagName("Keresztnev")
                                .item(i)
                                    .getTextContent());
                }

                System.out.println("Telefonszam : "
                    + eElement
                        .getElementsByTagName("Telefonszam")
                            .item(0)
                                .getTextContent());
                System.out.println("Eletkor : "
                    + eElement
                        .getElementsByTagName("Eletkor")
                            .item(0)
                                .getTextContent());
                System.out.println("Pozicio : "
                    + eElement
                        .getElementsByTagName("Pozicio")
                            .item(0)

```

```

        .getTextContent());
    }
}

private static void reszlegekListazasa(Document doc) {
    NodeList nList2 = doc.getElementsByTagName("Reszleg");
    NodeList menedzserLista = doc.getElementsByTagName("Menedzser");
    System.out.println("-----");

    for (int temp = 0; temp < nList2.getLength(); temp++) {
        Node nNode = nList2.item(temp);
        System.out.println("\nCurrent Element: " +
nNode.getNodeName());

        if (nNode.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement = (Element) nNode;
            System.out.println("ReszlegID : "
+ eElement.getAttribute("ReszlegID"));
            System.out.println("Vezeto : "
+ eElement.getAttribute("Vezeto"));
            System.out.println("Nev : "
+ eElement
.getElementsByTagName("Nev")
.item(0)
.getTextContent());
            {
                System.out.println("///Menedzser///");
                Node nNode2 = menedzserLista.item(temp);
                Element eElement2 = (Element) nNode2;

                System.out.println("MenedzserID : "
+ eElement2.getAttribute("MenedzserID"));
                System.out.println("Eletkor : "
+ eElement2
.getElementsByTagName("Eletkor")
.item(0)
.getTextContent());
                System.out.println("Fizetes : "
+ eElement2
.getElementsByTagName("Fizetes")
.item(0)
.getTextContent());

                System.out.println("Teljes nev : "
+ eElement2
.getElementsByTagName("Teljes_nev")
.item(0)
.getTextContent());

                System.out.println("Iranyitoszam : "
+ eElement2
.getElementsByTagName("Iranyitoszam")
.item(0)
.getTextContent());
                System.out.println("Orszag : "
+ eElement2
.getElementsByTagName("Orszag")
.item(0)
.getTextContent());
                System.out.println("Varos : "

```

```

        + eElement2
        .getElementsByTagName("Varos")
        .item(0)
        .getTextContent());
System.out.println("Utca : "
        + eElement2
        .getElementsByTagName("Utca")
        .item(0)
        .getTextContent());
System.out.println("Hazzsam : "
        + eElement2
        .getElementsByTagName("Hazzsam")
        .item(0)
        .getTextContent());
    }

    }

    }

    private static void alkalmazottakListazasa(Document doc) {
        NodeList alkalmazottLista =
doc.getElementsByTagName("Alkalmazott");
        System.out.println("-----");
        for (int temp = 0; temp < alkalmazottLista.getLength(); temp++) {
            Node nNode = alkalmazottLista.item(temp);
            System.out.println("\nCurrent Element : " +
nNode.getNodeName());

            if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                Element eElement = (Element) nNode;
                System.out.println("AlkalmazottID : "
                    + eElement.getAttribute("AlkalmazottID"));
                System.out.println("Reszleg : "
                    + eElement.getAttribute("Reszleg"));
                System.out.println("Nev : "
                    + eElement
                    .getElementsByTagName("Nev")
                    .item(0)
                    .getTextContent());

                System.out.println("Pozicio : "
                    + eElement
                    .getElementsByTagName("Pozicio")
                    .item(0)
                    .getTextContent());

                System.out.println("Kep helye : "
                    + eElement
                    .getElementsByTagName("Kep_helye")
                    .item(0)
                    .getTextContent());

                System.out.println("Keszites datuma : "
                    + eElement
                    .getElementsByTagName("Keszites_datuma")
                    .item(0)
                    .getTextContent());
            }
        }
    }
}

```

```

    }
}

private static void feltoltes(Document doc) throws TransformerException
{
    int flag = 0;

    Scanner sc = new Scanner(System.in);
    while(flag==0) {
        Node alkalmazottak =
doc.getElementsByTagName("Alkalmazottak").item(doc.getElementsByTagName("Al
kalmazottak").getLength() - 1);
        Element ujAlkalmazott = doc.createElement("Alkalmazott");
        Attr alkalmazottID = doc.createAttribute("AlkalmazottID");
        System.out.println("Adja meg az új alkalmazott egyedi
azonosítóját (1-nél nagyobb): ");
        String alkIDIn = sc.next();
        alkalmazottID.setValue(alkIDIn);

        Attr reszleg = doc.createAttribute("Reszleg");
        System.out.println("Adja meg az új alkalmazott részlegét (1-3):
");
        String alkReszlegIn = sc.next();
        reszleg.setValue(alkReszlegIn);

        ujAlkalmazott.setAttributeNode(alkalmazottID);
        ujAlkalmazott.setAttributeNode(reszleg);

        Element belepoKartya = doc.createElement("Belepokartya");

        Element alkalmazottNev = doc.createElement("Nev");
        System.out.println("Adja meg az új alkalmazott nevét: ");
        String alkNevIn = sc.next();
        alkalmazottNev.appendChild(doc.createTextNode(alkNevIn));

        Element alkalmazottPozicio = doc.createElement("Pozicio");
        System.out.println("Adja meg az új alkalmazott pozícióját: ");
        String alkPozIn = sc.next();
        alkalmazottPozicio.appendChild(doc.createTextNode(alkPozIn));

        Element belepoKep = doc.createElement("Kep");

        Element kepHelye = doc.createElement("Kep_helye");
        System.out.println("Adja meg az új alkalmazott képének elérési
útvonalát: ");
        String alkKepIn = sc.next();
        kepHelye.appendChild(doc.createTextNode(alkKepIn));

        Element keszitesDatuma = doc.createElement("Keszites_datuma");
        System.out.println("Adja meg az új alkalmazott kép készítésének
dátumát: ");
        String alkKepDatumIn = sc.next();
        keszitesDatuma.appendChild(doc.createTextNode(alkKepDatumIn));

        belepoKep.appendChild(kepHelye);
        belepoKep.appendChild(keszitesDatuma);

        belepoKartya.appendChild(alkalmazottNev);
        belepoKartya.appendChild(alkalmazottPozicio);
    }
}

```

```

        belepoKartya.appendChild(belepoKep);

        ujAlkalmazott.appendChild(belepoKartya);

        alkalmazottak.appendChild(ujAlkalmazott);

        System.out.println("Akar még több alkalmazottat felvinni? 0-
igen, különben nem");
        flag = sc.nextInt();
        if(flag!=0){
            break;
        }
    }

    // Tartalom írása XML fájlba
    TransformerFactory transformerFactory =
TransformerFactory.newInstance();
    Transformer transformer = transformerFactory.newTransformer();
    DOMSource source = new DOMSource(doc);

    ujFile(sc, transformer, source);

    /* Teszteléshez:
    StreamResult consoleResult = new StreamResult(System.out);
    transformer.transform(source, consoleResult);*/
}

private static void modosit(Document doc) throws TransformerException {
    Scanner sc = new Scanner(System.in);
    int input = 0;

    while(input < 1 || input > 3) {
        System.out.println("Az előző üzletév bevétele miatt egy részleg
dolgozóit előléptetjük.");

        System.out.println("Döntsön melyik részleg alkalmazottait
léptessük elő: ");
        System.out.println("1) Részleg1");
        System.out.println("2) Pénzügyi részleg");
        System.out.println("3) Műszaki részleg");
        input = sc.nextInt();
    }

    Node alkalmazottak =
doc.getElementsByTagName("Alkalmazottak").item(doc.getElementsByTagName("Al
kalmazottak").getLength() - 1);

    String eloleptetes = ""+input+"";

    NodeList list = alkalmazottak.getChildNodes();
    Node beosztas =
doc.getElementsByTagName("Pozicio").item(doc.getElementsByTagName("Pozicio"
).getLength() - 1);

    for (int temp = 0; temp < list.getLength(); temp++) {

```



```

        Node node = list.item(temp);
        if (node.getNodeType() == Node.ELEMENT_NODE) {
            Element eElement = (Element) node;
            if (eloleptetes.equals(eElement.getAttribute("Reszleg"))) {

                beosztas.setTextContent("Jól fizetett munkaerő");

            }
        }
    }

    // Tartalom kiírása a konzolba
    TransformerFactory transformerFactory =
TransformerFactory.newInstance();
    Transformer transformer = transformerFactory.newTransformer();
    DOMSource source = new DOMSource(doc);
    System.out.println("-----Módosított fájl-----");
    StreamResult consoleResult = new StreamResult(System.out);
    transformer.transform(source, consoleResult);

    //Tartalom kiírása fájlba
    ujFile(sc, transformer, source);
}

private static void ujFile(Scanner sc, Transformer transformer,
DOMSource source) throws TransformerException {
    System.out.println("");
    System.out.println("Adja meg az új XML fájl nevét kiterjesztés
nélkül: ");
    String newFile = sc.next();
    StreamResult result = new StreamResult(new File(newFile + ".xml"));
    transformer.transform(source, result);
    System.out.println("Új fájl létrehozva.");
}
}

```

Nyers XML fájl

```
<?xml version="1.0" encoding="UTF-8"?>
<Ceg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="file:/D:/xml-bead/beadando.xsd">
  <Vezetok>
    <Vezeto VezetoID="0">
      <Nevjegykartya>
        <E-Mail>ceo@company.com</E-Mail>
        <Vezeteknev>Tóth</Vezeteknev>
        <Keresztnev>János</Keresztnev>
        <Telefonszam>06701234567</Telefonszam>
      </Nevjegykartya>
      <Eletkor>50</Eletkor>
      <Pozicio>Vezérigazgató</Pozicio>
    </Vezeto>
    <Vezeto VezetoID="1">
      <Nevjegykartya>
        <E-Mail>importantguy@company.com</E-Mail>
        <Vezeteknev>Kovács</Vezeteknev>
        <Keresztnev>Ádám</Keresztnev>
        <Keresztnev>Zoltán</Keresztnev>
        <Telefonszam>06301234567</Telefonszam>
      </Nevjegykartya>
      <Eletkor>28</Eletkor>
      <Pozicio>Ügyvezető igazgató</Pozicio>
    </Vezeto>
    <Vezeto VezetoID="2">
      <Nevjegykartya>
        <E-Mail>valami@company.com</E-Mail>
        <Vezeteknev>Szabóné Szűcs</Vezeteknev>
        <Keresztnev>Éva</Keresztnev>
        <Telefonszam>06201234567</Telefonszam>
      </Nevjegykartya>
      <Eletkor>29</Eletkor>
      <Pozicio>Pénzügyi igazgató</Pozicio>
    </Vezeto>
    <Vezeto VezetoID="3">
      <Nevjegykartya>
        <E-Mail>admin@company.com</E-Mail>
        <Vezeteknev>Csabai</Vezeteknev>
        <Keresztnev>Zsolt</Keresztnev>
        <Telefonszam>06801234567</Telefonszam>
      </Nevjegykartya>
      <Eletkor>30</Eletkor>
      <Pozicio>Műszaki igazgató</Pozicio>
    </Vezeto>
  </Vezetok>
  <Reszlegek>
    <Reszleg ReszlegID="1" Vezeto="1">
      <Nev>Részleg1</Nev>
      <Menedzser MenedzserID="1">
        <Eletkor>21</Eletkor>
        <Fizetes>280000</Fizetes>
        <Teljes_nev>Tóth Péter</Teljes_nev>
        <Cim>
          <Iranyitoszam>1029</Iranyitoszam>
          <Orszag>Magyarország</Orszag>
          <Varos>Budapest</Varos>
          <Utca>Teszt</Utca>
          <Hazzsam>1</Hazzsam>
        </Cim>
      </Menedzser>
    </Reszleg>
  </Reszlegek>
</Ceg>
```

```

        </Menedzser>
    </Reszleg>
    <Reszleg ReszlegID="2" Vezeto="2">
        <Nev>Pénzügyi részleg</Nev>
        <Menedzser MenedzserID="11">
            <Eletkor>22</Eletkor>
            <Fizetes>290000</Fizetes>
            <Teljes_nev>Tóth Péter Károly</Teljes_nev>
            <Cim>
                <Iranyitoszam>1129</Iranyitoszam>
                <Ország>Magyarország</Ország>
                <Varos>Budapest</Varos>
                <Utca>Teszt</Utca>
                <Hazzsam>2</Hazzsam>
            </Cim>
        </Menedzser>
    </Reszleg>
    <Reszleg ReszlegID="3" Vezeto="3">
        <Nev>Műszaki részleg</Nev>
        <Menedzser MenedzserID="12">
            <Eletkor>23</Eletkor>
            <Fizetes>300000</Fizetes>
            <Teljes_nev>Tóth Péter László</Teljes_nev>
            <Cim>
                <Iranyitoszam>1229</Iranyitoszam>
                <Ország>Magyarország</Ország>
                <Varos>Budapest</Varos>
                <Utca>Teszt</Utca>
                <Hazzsam>3</Hazzsam>
            </Cim>
        </Menedzser>
    </Reszleg>
</Reszlegek>
<Alkalmazottak>
    <Alkalmazott AlkalmazottID="1" Reszleg="2">
        <Belepokartya>
            <Nev>Nagy Péter</Nev>
            <Pozicio>Munkásember</Pozicio>
            <Kep>
                <Kep_helye>https://company.com/images/private/nagypeter_20180101.jpg</Kep_helye>
                <Keszites_datuma>2018.01.01.</Keszites_datuma>
            </Kep>
        </Belepokartya>
    </Alkalmazott>
</Alkalmazottak>
</Ceg>

```