

# JEGYZŐKÖNYV

Adatkezelés XML környezetben

Féléves feladat

Társasjáték

Készítette: Szabó Martin

Neptun-kód: WJFAOO

Dátum: 2022.11.27.

# Tartalom

<b>A feladat leírása:</b> .....	3
<b>1.feladat</b> .....	4
<b>1.a) Az adatbázis ER modellje</b> .....	4
<b>1.b) Az adatbázis konvertálása XDM modellre</b> .....	4
<b>1.c) Az XDM modell alapján XML dokumentum készítése</b> .....	4
<b>1.d) Az XML dokumentum alapján XMLSchema készítése</b> .....	7
<b>2.Feladat</b> .....	10
<b>2.a) Adatolvasás (DomReadWJFAOO)</b> .....	11
<b>2.b) Adatlekérdezés (DOMQueryWJFAOO)</b> .....	12
<b>2.c) Adatmódosítás (DOMModifyWJFAOO)</b> .....	14

## A feladat leírása:

A feladatmegoldásom egy adott társasjáték lapjairól, valamint a lapokhoz tartozó tulajdonságokról készült. Az adatbázisban 5 egyed található, melyek eltérő számú tulajdonságokkal rendelkeznek.

A **Jatek\_nev**-hez 4 tulajdonság kapcsolódik. A **Jatek\_ID** a kulcs. A **kieg\_nev**, **nyelv** és a **kiad\_datum** pedig elemi tulajdonságok. Ezen egyenként a következő információkat hordozzák magukban:

- **Jatek\_ID** – ez a játék egyedi azonosítója/sorszáma.
- **kieg\_nev** – ez a játékban használt kiegészítő címe.
- **nyelv** – ez a tulajdonság jelzi azt, hogy milyen nyelvű a társas.
- **kiad\_datum** – a kiadás dátumát tudjuk meg belőle.

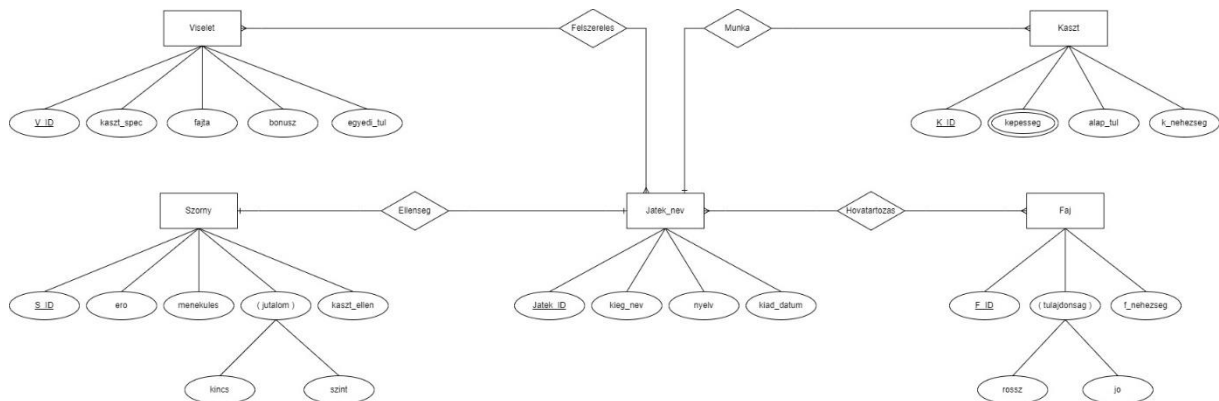
A **Jatek\_nev** egyedhez kapcsolódnak több-több kapcsolattal a **Viselet** és a **Faj** (Felszerelés és Hovatartozás), egy-több kapcsolattal a **Kaszt** (Munka) és egy-egy-el a **Szorny** (Ellenség) egyedek.

Ezek egyenként az alábbi tulajdonságokkal rendelkeznek:

- **Viselet:**
  - **V\_ID** – viselethez tartozó egyedi azonosító
  - **kaszt\_spec** – kasztokra vonatkozó egyedi tulajdonság
  - **fajta** – milyen típusú tárgyról van szó
  - **bonusz** – mennyi bónuszt ad
  - **egyedi\_tul** – valamilyen plusz tulajdonság
- **Faj:**
  - **F\_ID** – a fajhoz tartozó egyedi azonosító
  - **tulajdonsag** – adott faj tulajdonságai (jók és rosszak)
  - **jo** – pozitív kimenetelű ▪ **rossz** – negatív kimenetelű
  - **f\_nehezseg** – adott faj nehézsége
- **Kaszt:**
  - **K\_ID** – a kaszthoz tartozó egyedi azonosító
  - **kepesseg** – valamilyen egyedi képesség
  - **alap\_tul** – alapvető jellemzők
  - **k\_nehezseg** – a kaszt nehézsége
- **Szorny:**
  - **S\_ID** – a szörnyhöz tartozó egyedi azonosító
  - **ero** – milyen erős az adott lény
  - **menekules** – menekülés esetén mit vesztesz
  - **jutalom** – a legyőzésért járó jutalom (kincs és szint)
    - ✦ **kincs** – megszerzett kincs mennyisége
    - ✦ **szint** – megszerzett szint mennyisége
  - **kaszt\_ellen** – melyik kaszt ellen erősebb

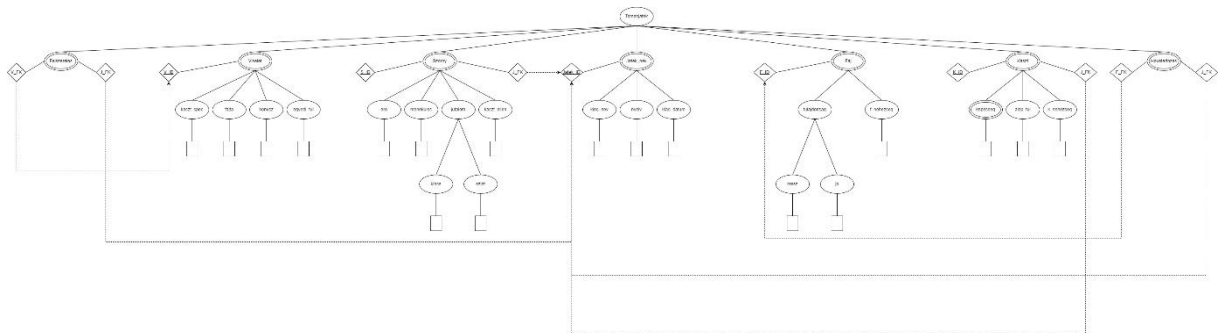
# 1.feladat

## 1.a) Az adatbázis ER modellje



## 1.b) Az adatbázis konvertálása XDM modellre

XDM modellnél háromféle jelölést szoktunk alkalmazni. Ezek az ellipszis, a rombusz, illetve a téglalap. Az ellipszis az elemeket jelöli, minden egyedből elem lesz, ezen felül a tulajdonságokból is. A rombusz az attribútumokat jelöli, amelyek a kulcs tulajdonságokból keletkeznek. A téglalap jelöli a szöveget, amely majd az XML dokumentumban fog megjelenni. Azoknak az elemeknek, amelyek többször is előfordulhatnak, a jelölése dupla ellipszissel történik. Az idegenkulcsok és a kulcsok közötti kapcsolatot szaggatott nyíllal jelöljük.



## 1.c) Az XDM modell alapján XML dokumentum készítése

```
<?xml version="1.0" encoding="UTF-8"?>

<Tarsasjatek xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xs:noNamespaceSchemaLocation="XMLSchemaWJFAOO.xsd">

  <!-- Jatekok-->
  <Jatek_nev Jatek_id="01">
    <kieg_nev>Zarandokok csatája</kieg_nev>
    <nyelv>Magyar</nyelv>
    <kiad_datum>2015.07.08.</kiad_datum>
  </Jatek_nev>

  <Jatek_nev Jatek_id="02">
    <kieg_nev>A harcos álma</kieg_nev>
    <nyelv>Magyar</nyelv>
```

```

    <kiad_datum>2017.12.01.</kiad_datum>
  </Jatek_nev>

  <Jatek_nev Jatek_id="03">
    <kieg_nev>A fáraó átka</kieg_nev>
    <nyelv></nyelv>
    <kiad_datum>2020.01.01.</kiad_datum>
  </Jatek_nev>

<!-- Szornyek -->
  <!-- Tavi Sejuani-->
  <Szorny S_id="01" S_J_id="01">
    <ero>Közepesen erős</ero>
    <menekules>-3 szint</menekules>
    <jutalom>
      <kins>+10 arany</kins>
      <szint>+5 szint</szint>
    </jutalom>
    <kaszt_ellen>Elf ellen erős</kaszt_ellen>
  </Szorny>
<!-- Ifjú Sárkányharcos-->
  <Szorny S_id="02" S_J_id="02">
    <ero>Közepesen erős</ero>
    <menekules>-3 szint</menekules>
    <jutalom>
      <kins>+10 arany</kins>
      <szint>+5 szint</szint>
    </jutalom>
    <kaszt_ellen>Ork ellen erős</kaszt_ellen>
  </Szorny>
<!-- Óriás Skorpió-->
  <Szorny S_id="03" S_J_id="03">
    <ero>Nagyon erős</ero>
    <menekules>-4 szint</menekules>
    <jutalom>
      <kins>+20 arany</kins>
      <szint>+7 szint</szint>
    </jutalom>
    <kaszt_ellen>Gnóm ellen erős</kaszt_ellen>
  </Szorny>

<!-- Kasztok -->
  <!--Bárd-->
  <Kaszt K_id="01" K_J_id="01">
    <kepesseg>+1 kincslap húzás</kepesseg>
    <alap_tul>3 lapért jutalom nélküli segítségkérés</alap_tul>
    <k_nehezseg>-1 szint harcok után a szintkapásból</k_nehezseg>
  </Kaszt>
  <!--Harcos-->
  <Kaszt K_id="02" K_J_id="02">
    <kepesseg>döntetlen harcok nyérése</kepesseg>
    <alap_tul>3 lapért +1 bónusz csatában</alap_tul>
    <k_nehezseg>kétszeres sebzést szenved el</k_nehezseg>
  </Kaszt>
  <!-- Tolvaj -->
  <Kaszt K_id="03" K_J_id="03">
    <kepesseg>1 lapért hátbaszúrás: -2 bónusz</kepesseg>

```

```

        <alap_tul>1 lapért tárgylopás</alap_tul>
        <k_nehezseg>Kevesebb arany harcok után</k_nehezseg>
    </Kaszt>

<!-- Fajok-->
<!-- Elf -->
<Faj F_id="01">
    <tulajdonsag>
        <rossz>-1 a meneküléshez</rossz>
        <jo>segítség esetén szintlépés</jo>
    </tulajdonsag>
    <f_nehezseg>-2 tárgycipelés</f_nehezseg>
</Faj>
<!-- Félszerzet -->
<Faj F_id="02">
    <tulajdonsag>
        <rossz>-1 a meneküléshez</rossz>
        <jo>dupla árú eladás</jo>
    </tulajdonsag>
    <f_nehezseg>-2 arany harcok után</f_nehezseg>
</Faj>
<!-- Gnóm -->
<Faj F_id="03">
    <tulajdonsag>
        <rossz>-1 a meneküléshez</rossz>
        <jo>szörny kijátszása magad mellett</jo>
    </tulajdonsag>
    <f_nehezseg>5 helyett csak 4 kártya</f_nehezseg>
</Faj>

<!-- Viseletek -->
<!-- Celebrimbör botja -->
<Viselet V_id="01">
    <kaszt_spec>Bárdoknak</kaszt_spec>
    <fajta>Elfeknek</fajta>
    <bonusz>+5 varázserő</bonusz>
    <egyedi_tul>+20% sebzés Szörnyek ellen</egyedi_tul>
</Viselet>
<!-- Jaksho sziklavértje -->
<Viselet V_id="02">
    <kaszt_spec>Harcosoknak</kaszt_spec>
    <fajta>Félszerzeteknek</fajta>
    <bonusz>+5 életerő</bonusz>
    <egyedi_tul>Kettővel kevesebb sebzést szenved el a viselő</egyedi_tul>
</Viselet>
<!-- A tolvajok kesztyűje -->
<Viselet V_id="03">
    <kaszt_spec>Tolvajoknak</kaszt_spec>
    <fajta>Gnómoknak</fajta>
    <bonusz>+5 sebesség</bonusz>
    <egyedi_tul>Minden sikeres lopás után +1 arany</egyedi_tul>
</Viselet>

<!-- Felszerelések (több-több kapcsolat) -->
<Felszereles F_V_id="01" F_J_id="01"></Felszereles>

<Felszereles F_V_id="02" F_J_id="02"></Felszereles>

```

```

<Felszerelés F_V_id="03" F_J_id="03"></Felszerelés>

<!-- Hovatartozások (több-több kapcsolat) -->
<Hovatartozás H_F_id="01" H_J_id="01"></Hovatartozás>

<Hovatartozás H_F_id="02" H_J_id="02"></Hovatartozás>

<Hovatartozás H_F_id="03" H_J_id="03"></Hovatartozás>

</Társasjáték>

```

#### 1.d) Az XML dokumentum alapján XMLSchema készítése

Az XML dokumentumhoz validálásához kellett egy sémát létrehozni. Először is az egyszerű típusokat definiáltam, melyekre majd a ref kulcsszóval fogok hivatkozni a sémán belül. Ezután az attribútumokat definiáltam, majd az összetett típusokat, így sokkal áttekinthetőbb lett maga a séma. A kód, kommentekkel ellátva, az alábbiakban megtekinthető.

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
<!-- Egyszerű típusok definiálása -->
  <!--Jatek_nev egyszerű típusai -->
  <xs:element name="kieg_nev" type="xs:string"></xs:element>
  <xs:element name="nyelv" type="xs:string"></xs:element>
  <xs:element name="kiad_datum" type="xs:string"></xs:element>

  <!-- Szornyek egyszerű típusai -->
  <xs:element name="ero" type="xs:string"></xs:element>
  <xs:element name="menekules" type="xs:string"></xs:element>
  <xs:element name="kincs" type="xs:string"></xs:element>
  <xs:element name="szint" type="xs:string"></xs:element>
  <xs:element name="kaszt_ellen" type="xs:string"></xs:element>

  <!-- Kasztok egyszerű típusai -->
  <xs:element name="kepesseg" type="xs:string"></xs:element>
  <xs:element name="alap_tul" type="xs:string"></xs:element>
  <xs:element name="k_nehezseg" type="xs:string"></xs:element>

  <!-- Kasztok egyszerű típusai -->
  <xs:element name="rossz" type="xs:string"></xs:element>
  <xs:element name="jo" type="xs:string"></xs:element>
  <xs:element name="f_nehezseg" type="xs:string"></xs:element>

  <!-- Viseletek egyszerű típusai -->
  <xs:element name="kaszt_spec" type="xs:string"></xs:element>
  <xs:element name="fajta" type="xs:string"></xs:element>
  <xs:element name="bonusz" type="xs:string"></xs:element>
  <xs:element name="egyedi_tul" type="xs:string"></xs:element>

<!-- Attributumok definiálása -->
  <xs:attribute name="Jatek_id" type="xs:string" />
  <xs:attribute name="S_id" type="xs:string" />
  <xs:attribute name="S_J_id" type="xs:string" />

```

```

<xs:attribute name="K_id" type="xs:string" />
<xs:attribute name="K_J_id" type="xs:string" />
<xs:attribute name="F_id" type="xs:string" />
<xs:attribute name="V_id" type="xs:string" />
<xs:attribute name="F_V_id" type="xs:string" />
<xs:attribute name="F_J_id" type="xs:string" />
<xs:attribute name="H_F_id" type="xs:string" />
<xs:attribute name="H_J_id" type="xs:string" />

<!-- Összetett típusok definiálása -->
<!-- Jatek_nev típus -->
<xs:complexType name="Jatek_nev_tipus">
  <xs:sequence>
    <xs:element ref="kieg_nev"></xs:element>
    <xs:element ref="nyelv"></xs:element>
    <xs:element ref="kiad_datum"></xs:element>
  </xs:sequence>
  <xs:attribute ref="Jatek_id" use="required"></xs:attribute>
</xs:complexType>

<!-- Szorny típus -->
<xs:complexType name="Szorny_tipus">
  <xs:sequence>
    <xs:element ref="ero"></xs:element>
    <xs:element ref="menekules"></xs:element>
    <xs:element name="jutalom">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="kinscs"></xs:element>
          <xs:element ref="szint"></xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element ref="kaszt_ellen"></xs:element>
  </xs:sequence>
  <xs:attribute ref="S_id" use="required"></xs:attribute>
  <xs:attribute ref="S_J_id" use="required"></xs:attribute>
</xs:complexType>

<!-- Kaszt típus -->
<xs:complexType name="Kaszt_tipus">
  <xs:sequence>
    <xs:element ref="kepesseg"></xs:element>
    <xs:element ref="alap_tul"></xs:element>
    <xs:element ref="k_nehezseg"></xs:element>
  </xs:sequence>
  <xs:attribute ref="K_id" use="required"></xs:attribute>
  <xs:attribute ref="K_J_id" use="required"></xs:attribute>
</xs:complexType>

<!-- Faj típus -->
<xs:complexType name="Faj_tipus">
  <xs:sequence>
    <xs:element name="tulajdonsag">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="rossz"></xs:element>

```



```

        <xs:element ref="jo"></xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element ref="f_nehezseg"></xs:element>
</xs:sequence>
<xs:attribute ref="F_id" use="required"></xs:attribute>
</xs:complexType>

<!-- Viselet típus -->
<xs:complexType name="Viselet_tipus">
    <xs:sequence>
        <xs:element ref="kaszt_spec"></xs:element>
        <xs:element ref="fajta"></xs:element>
        <xs:element ref="bonusz"></xs:element>
        <xs:element ref="egyedi_tul"></xs:element>
    </xs:sequence>
    <xs:attribute ref="V_id" use="required"></xs:attribute>
</xs:complexType>

<xs:complexType name="Felszerelés_tipus">
    <xs:attribute ref="F_V_id" use="required"></xs:attribute>
    <xs:attribute ref="F_J_id" use="required"></xs:attribute>
</xs:complexType>

<xs:complexType name="Hovatartozás_tipus">
    <xs:attribute ref="H_F_id" use="required"></xs:attribute>
    <xs:attribute ref="H_J_id" use="required"></xs:attribute>
</xs:complexType>

<!-- Maga a séma -->
<xs:element name="Tarsasjatek">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Jatek_nev" type="Jatek_nev_tipus"
maxOccurs="unbounded" />
            <xs:element name="Szorny" type="Szorny_tipus" maxOccurs="unbounded"
/>
            <xs:element name="Kaszt" type="Kaszt_tipus" maxOccurs="unbounded" />
            <xs:element name="Faj" type="Faj_tipus" maxOccurs="unbounded" />
            <xs:element name="Viselet" type="Viselet_tipus" maxOccurs="unbounded"
/>
            <xs:element name="Felszerelés" type="Felszerelés_tipus"
maxOccurs="unbounded"/>
            <xs:element name="Hovatartozás" type="Hovatartozás_tipus"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>

    <!-- Kulcsok -->
    <xs:key name="Jatek_nev_kulcs">
        <xs:selector xpath="Jatek_nev"></xs:selector>
        <xs:field xpath="@Jatek_id"></xs:field>
    </xs:key>

    <xs:key name="Szorny_kulcs">
        <xs:selector xpath="Szorny"></xs:selector>

```

```

    <xs:field xpath="@S_id"></xs:field>
  </xs:key>

  <xs:key name="Kaszt_kulcs">
    <xs:selector xpath="Kaszt"></xs:selector>
    <xs:field xpath="@K_id"></xs:field>
  </xs:key>

  <xs:key name="Faj_kulcs">
    <xs:selector xpath="Faj"></xs:selector>
    <xs:field xpath="@F_id"></xs:field>
  </xs:key>

  <xs:key name="Viselet_kulcs">
    <xs:selector xpath="Viselet"></xs:selector>
    <xs:field xpath="@V_id"></xs:field>
  </xs:key>

  <!-- Idegen kulcsok -->
  <xs:keyref name="Kaszt_idegen_kulcs" refer="Jatek_nev_kulcs">
    <xs:selector xpath="Kaszt"></xs:selector>
    <xs:field xpath="@K_J_id"></xs:field>
  </xs:keyref>

  <xs:keyref name="Faj_idegen_kulcs" refer="Faj_kulcs">
    <xs:selector xpath="Hovatartozas"></xs:selector>
    <xs:field xpath="@H_F_id"></xs:field>
  </xs:keyref>

  <xs:keyref name="Viselet_idegen_kulcs" refer="Viselet_kulcs">
    <xs:selector xpath="Felszerelés"></xs:selector>
    <xs:field xpath="@F_V_id"></xs:field>
  </xs:keyref>

  <xs:keyref name="Felszerelés_idegen_kulcs" refer="Jatek_nev_kulcs">
    <xs:selector xpath="Felszerelés"></xs:selector>
    <xs:field xpath="@F_J_id"></xs:field>
  </xs:keyref>

  <xs:keyref name="Hovatartozas_idegen_kulcs" refer="Jatek_nev_kulcs">
    <xs:selector xpath="Hovatartozas"></xs:selector>
    <xs:field xpath="@H_J_id"></xs:field>
  </xs:keyref>

  <!-- 1:1 kapcsolat megvalósítása -->
  <xs:unique name="unique_Szorny">
    <xs:selector xpath="Szorny"></xs:selector>
    <xs:field xpath="@S_J_id"></xs:field>
  </xs:unique>
</xs:element>
</xs:schema>

```

## 2.Feladat

A feladat egy DOM program készítése az XML dokumentum – XMLWjfaoo.xml – adatainak adminisztrálása alapján.

## 2.a) Adatolvasás (DomReadWJFAOO)

A Jatek\_nev egyed adatait fogom kiírni ezzel a DOM programmal.

A kommentekkel ellátott kód:

```
package hu.domparse.wjfaoo;

import java.io.File;
import java.io.IOException;

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

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;

public class DomReadWJFAOO {

    public static void main(String[] args) throws ParserConfigurationException,
SAXException, IOException {
        //Forrás file
        File file = new File("XMLWjfaoo.xml");

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        DocumentBuilder dBuilder = factory.newDocumentBuilder();

        Document doc = dBuilder.parse(file);

        doc.getDocumentElement().normalize();
        //Gyökérelem
        System.out.println("Root element: " +
doc.getDocumentElement().getNodeName());
        //Gyerekelemek lementése
        NodeList nList = (NodeList) doc.getDocumentElement();

        for (int i = 0; i < nList.getLength(); i++) {
            Node node = nList.item(i);

            //Ha Jatek_nev
            if (node.getNodeName() == "Jatek_nev") {
                if (!node.getNodeName().equals("#text")) {
                    System.out.println("\n");
                    System.out.println("Current element: " + node.getNodeName());
                }
                //Jatek_nev adatainak kiírása
                if (node.getNodeType() == Node.ELEMENT_NODE) {
                    Element elem = (Element) node;

                    String jatek_id = elem.getAttribute("Jatek_id");

                    Node jatek_node = elem.getElementsByTagName("kieg_nev").item(0);
```

```

        String kieg_nev_name = jatek_node.getTextContent();

        Node jatek_node2 = elem.getElementsByTagName("nyelv").item(0);
        String nyelv_name = jatek_node2.getTextContent();

        Node jatek_node3 = elem.getElementsByTagName("kiad_datum").item(0);
        String kiad_datum_name = jatek_node3.getTextContent();

        System.out.printf("Jatek id: %s%n", jatek_id);
        System.out.printf("Kiegeszito neve: %s%n", kieg_nev_name);
        System.out.printf("A jatek nyelve: %s%n", nyelv_name);
        System.out.printf("Kiadas datuma: %s%n", kiad_datum_name);
    }
}
}
}
}

```

## 2.b) Adatlekérdezés (DOMQueryWJFAOO)

Az adatok lekérdezését az XPath használatával készítettem el, 6 lekérdezést hajtottam végre, és ezeket ki is írtam a konzolra.

A 6 lekérdezés:

- A Tarsasjatek (Root elem) Kaszt gyerekelemeinek a lekérdezése
- 3-as ID-jű Viselet lekérdezése
- A második Jatek\_nev kiválasztása
- Annak a kasztnak a kiírása, melynek a képessége: +1 kincslap húzás
- Jatek\_nev egyed kiegyező neve, kiadási dátuma
- Azok a kasztok, amelyeknek van attribútuma

A kommentekkel ellátott kód:

```

package hu.domparse.wjfaoo;
import java.io.IOException;

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 org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;
public class DOMQueryWJFAOO {
    public static void main(String[] args) {

```

```

    try {
// DocumentBuilder létrehozása
        DocumentBuilderFactory documentBuilderFactory =
            DocumentBuilderFactory.newInstance();
        DocumentBuilder documentBuilder =
            documentBuilderFactory.newDocumentBuilder();
        Document document =
            documentBuilder.parse("XMLWjfaoo.xml");
        document.getDocumentElement().normalize();
// az XPath készítése
        XPath xPath =
            XPathFactory.newInstance().newXPath();
// meg kell adni az elérési út kifejezést és csomópont listát
// a Tarsasjatek (Root elem) Kaszt gyerekelemeinek lekérdezése

        String expression = "Tarsasjatek / Kaszt";
// 3-as ID-jű Viselet lekérdezése
//String expression = "//Viselet[@V_id='03']";
// a második Jatek_nev kiválasztása
//String expression = "Tarsasjatek/Jatek_nev[2]";
// az a kaszt, aminek a képessége : + 1 kincslap húzás
//String expression = "//Kaszt[kepesseg='+1 kincslap húzás']";
// Jatek_nev kiegészítő neve, és kiadási dátuma
//String expression = "//kieg_nev | //kiad_datum";
//Kasztok akiknek van attribútuma
//String expression = "//Kaszt[@*]";
// Listakészítés, xPath kifejezés ----> fordítás + értékelés
        NodeList nodeList = (NodeList)
            xPath.compile(expression).evaluate(document,
                XPathConstants.NODESET);
// A NodeList csomópontjain való iterálás
        for (int i = 0; i < nodeList.getLength(); i++) {
            Node node = nodeList.item(i);
            System.out.println("\nAktuális elem: " + node.getNodeName());
// Csomópontvizsgálás, Subelement tesztelése (most Kaszt)
            if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("Kaszt")) {
                Element element = (Element) node;
                System.out.println("ID: " + element.getAttribute("K_id"));
                System.out.println("Képesség: " +
element.getElementsByTagName("kepesseg").item(0).getTextContent());
                System.out.println("Alap tulajdonság: " +
element.getElementsByTagName("alap_tul").item(0).getTextContent());
                System.out.println("Kaszt nehézsége: " +
element.getElementsByTagName("k_nehezseg").item(0).getTextContent());
            }
//Jatek_nev kiírása
            if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("Jatek_nev")) {
                Element element = (Element) node;
                System.out.println("ID: " + element.getAttribute("Jatek_id"));
                System.out.println("Kiegészítő neve: " +
element.getElementsByTagName("kieg_nev").item(0).getTextContent());
                System.out.println("A játék nyelve: " +
element.getElementsByTagName("nyelv").item(0).getTextContent());
                System.out.println("A játék kiadási dátuma: " +
element.getElementsByTagName("kiad_datum").item(0).getTextContent());
            }
        }
    }
}

```

```

    }
    //Viselet kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("Viselet")) {
        Element element = (Element) node;
        System.out.println("ID: " + element.getAttribute("V_id"));
        System.out.println("Kaszt neve, melynek speciális felszerelése: " +
element.getElementsByTagName("kaszt_spec").item(0).getTextContent());
        System.out.println("Milyen fajnak a felszerelése: " +
element.getElementsByTagName("fajta").item(0).getTextContent());
        System.out.println("Milyen bónuszt ad: " +
element.getElementsByTagName("bonusz").item(0).getTextContent());
        System.out.println("Egyedi tulajdonság: " +
element.getElementsByTagName("egyedi_tul").item(0).getTextContent());
    }
    //Kaszt képességének kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("kepesseg")) {
        Element element = (Element) node;
        System.out.println("A kaszt képessége: " + element.getTextContent());
    }
    // Jatek_Nev kiegeg_nev kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("kieg_nev")) {
        Element element = (Element) node;
        System.out.println("A játék kiegészítőjének a neve: " +
element.getTextContent());
    }
    // Jatek_nev kiad_datum kiírása
    if (node.getNodeType() == Node.ELEMENT_NODE &&
node.getNodeName().equals("kiad_datum")) {
        Element element = (Element) node;
        System.out.println("Kiadási dátum: " + element.getTextContent());
    }
}
} catch (ParserConfigurationException e) {
    e.printStackTrace();
} catch (SAXException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
} catch (XPathExpressionException e) {
    e.printStackTrace();
}
}
}

```

## 2.c) Adatmódosítás (DOMModifyWJFAOO)

Az első és a harmadik Kaszt egyedet elmentem.

Adatmódosítások, amelyek megtörténtek:

- Harmadik Kasztnak az ID váltása

- Első Kaszt képességének megváltoztatása
- Első Kaszt alaptulajdonságának a megváltoztatása
- A Faj egyedek törlése

A kommentekkel ellátott kód:

```
package hu.domp.parse.wjfaoo;
import java.io.File;
import java.io.IOException;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;

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

public class DOMModifyWJFAOO {
    public static void main(String[] args) throws ParserConfigurationException,
        SAXException, IOException {

        try {
            //Forrás file
            File inputFile = new File("XML2Wjfaoo.xml");
            DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
            DocumentBuilder docBuilder = docFactory.newDocumentBuilder();
            Document doc = docBuilder.parse(inputFile);

            //Első és harmadik Kaszt mentése
            Node kaszt1 = doc.getElementsByTagName("Kaszt").item(0);
            Node kaszt3 = doc.getElementsByTagName("Kaszt").item(2);
            //Gyökérelem
            Node Tarsasjatek = doc.getFirstChild();

            //Harmadik Kasztnak az ID váltása
            NamedNodeMap attr = kaszt3.getAttributes();
            Node nodeAttr = attr.getNamedItem("K_id");
            nodeAttr.setTextContent("06");

            //Első Kaszt képességének megváltoztatása
            NodeList list = kaszt1.getChildNodes();
            for (int i = 0; i < list.getLength(); i++) {
                Node node = list.item(i);
                if (node.getNodeType() == Node.ELEMENT_NODE) {
                    Element elem = (Element) node;
                    if ("kepesseg".equals(elem.getNodeName())) {
                        elem.setTextContent("nincs képessége");
                    }
                }
            }
        }
    }
}
```

```

    }

    //Első Kasztnak az alap tulajdonságának a megváltoztatása
    NodeList list1 = kaszt1.getChildNodes();
    for (int i = 0; i < list1.getLength(); i++) {
        Node node1 = list1.item(i);
        if (node1.getNodeType() == Node.ELEMENT_NODE) {
            Element elem1 = (Element) node1;
            if ("alap_tul".equals(elem1.getNodeName())) {
                elem1.setTextContent("nincs alap tulajdonsága");
            }
        }
    }
}

//Fajok törlése
NodeList childNodes = Tarsasjatek.getChildNodes();
for (int i = 0; i < childNodes.getLength(); i++) {
    Node node = childNodes.item(i);

    if ("Faj".equals(node.getNodeName()))
        Tarsasjatek.removeChild(node);
}

//Konzolra kérés
TransformerFactory transformerFactory = TransformerFactory.newInstance();
Transformer transformer = transformerFactory.newTransformer();
DOMSource source = new DOMSource(doc);
System.out.println("New File");
StreamResult consoleResult = new StreamResult(System.out);
transformer.transform(source, consoleResult);
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```