

JEGYZŐKÖNYV

Rally versenyekről szóló adatok felépítése

XML-ben

Adatkezelés XML környezetben
Féléves feladat

Készítette: **Szabó Dániel**
Neptunkód: **XUXEJO**

Feladat leírása:

- Az általam megtervezett és létrehozott XML Rally Versenyekről fog szólni. Tárolja az adott Szakasz adatait és annak versenyzőit. Számon tartja a Versenyzők nemzetiségét. Szakaszokon lement verseny Autókat, pilótáik nemzetiségét. Tárolja hogy az adott Szakasz melyik Országban van. Ezek alapján könnyen le lehetkövetni az adott Rallynak az alapvető információit (Autók, Versenyzők, Nemzetiségük, stb.). Az adott Szakaszon milyen hőmérsékletek uralkodnak, milyen hosszú és mi a versenyszakasz felülete, legyen az murva, aszfalt, durva kavicsos murva, hó, stb. Versenyzők melyik autót vezetik, milyen időket mentek a szakaszokon és még sok más.

1. Feladat:

a) Az XML adatstruktúrának az ER modellje - Az ER modell egyedei és tulajdonságai:

- **Egyedek:** Szakaszz, Versenyző, Nemzetiség, Autó, Ország

- **Szakasz:**

Név
Hőmérséklet
Hossz
Felület

- **Versenyző:**

VKód Név

- **Nemzetiség:**

NemzetKód

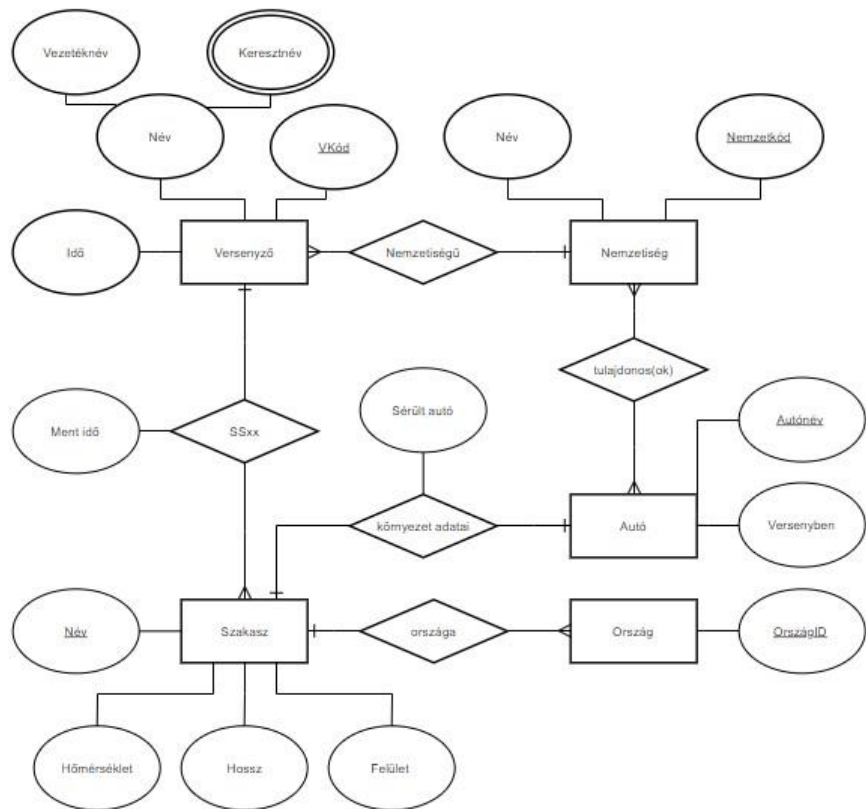
Név - **Autó:**

AutóNév

Versenyben

- **Ország:**

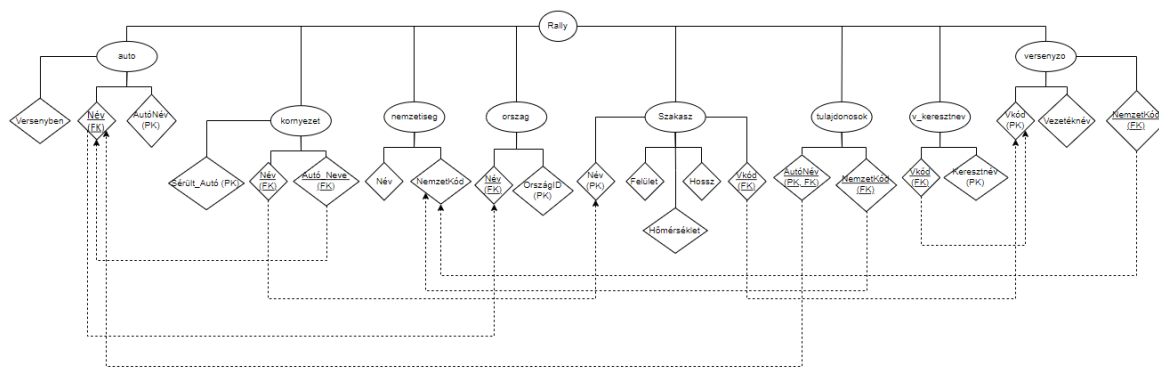
OrszágID



Egyedek közötti kapcsolat:

- A **Szakasz** és a **Versenyző** között 1:N típusú kapcsolat van, mivel egy versenyző több szakaszon is részt vehet.
- A **Versenyző** és a **Nemzetiség** között 1:N típusú kapcsolat van, mivel egy nemzethez tartozhat több versenyző, de egy versenyzőnek nem lehet több nemzetisége.
- Az **Autó** és a **Nemzetiség** között N:M típusú kapcsolat van, mivel egy autója több nemzethez is lehet és egy nemzethez lehet több autója is.
- A **Szakasz** és az **Autó** között 1:1 típusú kapcsolat van.
- Az **Ország** és a **Szakasz** között 1:N típusú kapcsolat van, mivel egy országnak lehet több szakasza, de egy adott szakasz csak egy országban lehet.

c) Az XML adatstruktúrának az XDM modellje



- **Root Element, azaz a gyökér elem:** Rally

- A gyökér elemhez tartozó node-ok: auto, kornyezet, nemzetiseg, ország, szakasz, tulajdonos, v_keresztnev, versenyzo

- Mindegyik node különböző adatokat tartalmaz;

- **Auto:** AutóNév, Versenyben van-e még a gépjármű, illetve a Szakasznak a Neve
- **Kornyezet:** Igen/Nem értékkel válaszolható, hogy sérült-e a szakaszon az auto, Mi az auto Neve, és mi a szakasz neve
- **Nemzetiseg:** Nemzetkód sima Integer értékű, nemzetiség elnevezése
- **Ország:** OrszágID mely úgyszintén egy Integer érték, És a szakasz_neve, hogy melyik országhoz tartozik
- **Szakasz:** Szakasz különböző tulajdonságai, Hőmérséklet, Szakasz hossz, Felület (murva, aszfalt, stb.), stb...
- **Tulajdonos:** AutóNév és a NemzetKód, hogy meg lehessen állapítani, hogy kihez tartozik az auto, melyik versenyzőhöz
- **V_keresztnev:** Versenyzők keresztnéve ID-vel (VKód) ellátva
- **Versenyzo:** Versenyzők ID-je, vezetéknéve és nemzetisége található benne

d) Az XML fájl tartalma:

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

<Rally>
  <!-- Tábla Autó -->
  <auto id="1" brand="bmw">
    <AutóNév type="murvára">bmw_m3_e30</AutóNév>
    <Versenyben>igen</Versenyben>
    <Név>magyar_szakaszh</Név>
  </auto>

  <auto id="2" brand="ford">
    <AutóNév type="murvára">ford_fiesta_r5</AutóNév>
    <Versenyben>nem</Versenyben>
    <Név type="murva">olaszh_szakaszh</Név>
  </auto>

  <auto id="3" brand="ford">
    <AutóNév type="aszfaltra">ford_focus_wrc</AutóNév>
    <Versenyben>nem</Versenyben>
    <Név>francia_szakaszh</Név>
  </auto>

  <auto id="4" brand="skoda">
    <AutóNév type="aszfaltra">skoda_fabia_r5</AutóNév>
    <Versenyben>nem</Versenyben>
    <Név>orosz_szakaszh</Név>
  </auto>

  <auto id="5" brand="volkswagen">
    <AutóNév type="Vegyes">wv_polo_r5</AutóNév>
    <Versenyben>igen</Versenyben>
    <Név>nemet_szakaszh</Név>
  </auto>

  <!-- Tábla Környezet -->
  <kornyezh id="1">
    <Sérült_autó>nem</Sérült_autó>
    <Név>francia_szakaszh</Név>
    <Autó_Neve>ford_focus_wrc</Autó_Neve>
  </kornyezh>

  <kornyezh id="2">
    <Sérült_autó>igen</Sérült_autó>
    <Név>nemet_szakaszh</Név>
    <Autó_Neve>wv_polo_r5</Autó_Neve>
  </kornyezh>

  <kornyezh id="3">
    <Sérült_autó>igen</Sérült_autó>
    <Név>magyar_szakaszh</Név>
    <Autó_Neve>bmw_m3_e30</Autó_Neve>
  </kornyezh>

  <kornyezh id="4">
    <Sérült_autó>nem</Sérült_autó>
    <Név>olaszh_szakaszh</Név>
    <Autó_Neve>ford_fiesta_r5</Autó_Neve>
  </kornyezh>
</Rally>
```

```

<kornyezet id="5">
  <Sérült_autó>nem</Sérült_autó>
  <Név>orosz_szakasz</Név>
  <Autó Neve>skoda fabia r5</Autó Neve>
</kornyezet>

<!-- Tábla Nemzetiség -->
<nemzetiseg id="1">
  <NemzetKód>1</NemzetKód>
  <Név>cseh</Név>
</nemzetiseg>

<nemzetiseg id="2">
  <NemzetKód>2</NemzetKód>
  <Név>francia</Név>
</nemzetiseg>

<nemzetiseg id="3">
  <NemzetKód>3</NemzetKód>
  <Név>német</Név>
</nemzetiseg>

<nemzetiseg id="4">
  <NemzetKód>4</NemzetKód>
  <Név>olasz</Név>
</nemzetiseg>

<nemzetiseg id="5">
  <NemzetKód>5</NemzetKód>
  <Név>magyar</Név>
</nemzetiseg>

<!-- Tábla Ország -->
<ország id="1">
  <OrszágID>3</OrszágID>
  <Név>francia_szakasz</Név>
</ország>

<ország id="2">
  <OrszágID>1</OrszágID>
  <Név>magyar_szakasz</Név>
</ország>

<ország id="3">
  <OrszágID>4</OrszágID>
  <Név>nemet_szakasz</Név>
</ország>

<ország id="4">
  <OrszágID>2</OrszágID>
  <Név>olasz_szakasz</Név>
</ország>

<ország id="5">
  <OrszágID>5</OrszágID>
  <Név>orosz_szakasz</Név>
</ország>

<!-- Tábla Szakasz -->
<szakasz id="1">
  <Név>francia_szakasz</Név>

```

```
<Hőmérséklet>15</Hőmérséklet>
<Hossz>12</Hossz>
<Felület>aszfalt</Felület>
<Vkód>5</Vkód>
</szakasz>

<szakasz id="2">
  <Név>magyar_szakasz</Név>
  <Hőmérséklet>12</Hőmérséklet>
  <Hossz>25</Hossz>
  <Felület>murva</Felület>
  <Vkód>1</Vkód>
</szakasz>

<szakasz id="3">
  <Név>nemet_szakasz</Név>
  <Hőmérséklet>4</Hőmérséklet>
  <Hossz>25</Hossz>
  <Felület>aszfalt-murva</Felület>
  <Vkód>4</Vkód>
</szakasz>

<szakasz id="4">
  <Név>olasz_szakasz</Név>
  <Hőmérséklet>9</Hőmérséklet>
  <Hossz>30</Hossz>
  <Felület>murva</Felület>
  <Vkód>3</Vkód>
</szakasz>

<szakasz id="5">
  <Név>orosz_szakasz</Név>
  <Hőmérséklet>-1</Hőmérséklet>
  <Hossz>27</Hossz>
  <Felület>aszfalt</Felület>
  <Vkód>2</Vkód>
</szakasz>

<!-- Tábla Tulajdonos -->
<tulajdonos id="1">
  <AutóNév>bmw_me_e30</AutóNév>
  <NemzetKód>5</NemzetKód>
</tulajdonos>

<tulajdonos id="2">
  <AutóNév>ford_fiesta_r5</AutóNév>
  <NemzetKód>2</NemzetKód>
</tulajdonos>

<tulajdonos id="3">
  <AutóNév>ford_focus_wrc</AutóNév>
  <NemzetKód>3</NemzetKód>
</tulajdonos>

<tulajdonos id="4">
  <AutóNév>skoda_fabia_r5</AutóNév>
  <NemzetKód>4</NemzetKód>
</tulajdonos>

<tulajdonos id="5">
  <AutóNév>vw_polo_r5</AutóNév>
```

```

        <NemzetKód>1</NemzetKód>
    </tulajdonos>

    <!-- Tábla Versenyző -->
    <versenyzo id="1">
        <Vkód>1</Vkód>
        <Vezetéknév>Tanak</Vezetéknév>
        <Nemzetkód>5</Nemzetkód>
    </versenyzo>

    <versenyzo id="2">
        <Vkód>2</Vkód>
        <Vezetéknév>Ogier</Vezetéknév>
        <Nemzetkód>3</Nemzetkód>
    </versenyzo>

    <versenyzo id="3">
        <Vkód>3</Vkód>
        <Vezetéknév>Rovenpera</Vezetéknév>
        <Nemzetkód>1</Nemzetkód>
    </versenyzo>

    <versenyzo id="4">
        <Vkód>4</Vkód>
        <Vezetéknév>Neuville</Vezetéknév>
        <Nemzetkód>2</Nemzetkód>
    </versenyzo>

    <versenyzo id="5">
        <Vkód>5</Vkód>
        <Vezetéknév>Perez</Vezetéknév>
        <Nemzetkód>4</Nemzetkód>
    </versenyzo>

    <!-- Tábla V_Keresztnév -->
    <v_keresztnev id="1">
        <Keresztnév>Ott</Keresztnév>
        <Vkód>1</Vkód>
    </v_keresztnev>

    <v_keresztnev id="2">
        <Keresztnév>Sebastian</Keresztnév>
        <Vkód>2</Vkód>
    </v_keresztnev>

    <v_keresztnev id="3">
        <Keresztnév>Kella</Keresztnév>
        <Vkód>3</Vkód>
    </v_keresztnev>

    <v_keresztnev id="4">
        <Keresztnév>Thierry</Keresztnév>
        <Vkód>4</Vkód>
    </v_keresztnev>

    <v_keresztnev id="5">
        <Keresztnév>Segio</Keresztnév>
        <Vkód>5</Vkód>
    </v_keresztnev>
</Rally>

```


d) Az XML Schema fájl tartalma:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- autó -->
  <xs:element name="rally">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="autóTipusok"
type="autoTipus"/></xs:element>
        <xs:element name="országok"
type="országTipus"/></xs:element>
        <xs:element name="nemzetisegek"
type="nemzetisegTipus"/></xs:element>
        <xs:element name="kornyezet"
type="kornyezetTipus"/></xs:element>
        <xs:element name="szakaszJellmezés"
type="szakaszJellemTipus"/></xs:element>
        <xs:element name="AutóTulajdonos"
type="tulajdonTipus"/></xs:element>
        <xs:element name="VersenyzőVezeték"
type="versenyzoVezetekTipus"/></xs:element>
        <xs:element name="VersenyzőKereszt"
type="versenyzoKeresztTipus"/></xs:element>
      </xs:sequence>
    </xs:complexType>
  <!-- Környezet Tipusai -->
  <xs:key name="kornyezetPKSzakasz">
    <xs:selector xpath="kornyezet"/>
    <xs:field xpath="@Szakasz"/>
  </xs:key>

  <xs:keyref refer="kornyezetPKSzakasz" name="szakaszFK">
    <xs:selector xpath="rally/szakaszJellmezés"/>
    <xs:field xpath="@Szakasz"/>
  </xs:keyref>

  <!-- Szakasz -->
  <xs:key name="szakaszJellmezésPK">
    <xs:selector xpath="szakaszJellmezés"/>
    <xs:field xpath="@Vkód"/>
  </xs:key>

  <xs:keyref refer="szakaszJellmezésPK" name="versenyzoFK">
    <xs:selector xpath="rally/VersenyzőVezeték"/>
    <xs:field xpath="@Vkód"/>
  </xs:keyref>

  <!-- Versenyző keresztnéve -->
  <xs:key name="vezeteknevPK">
    <xs:selector xpath="VersenyzőKereszt"/>
    <xs:field xpath="@Vkód"/>
  </xs:key>

  <xs:keyref refer="vezeteknevPK" name="versenyzoKeresztFK">
    <xs:selector xpath="rally/VersenyzőKereszt"/>
    <xs:field xpath="@Vkód"/>
  </xs:keyref>

  <!-- Versenyző PK -->
  <xs:key name="vezeteknevNemzetkodPK">
```

```

        <xs:selector xpath="VersenyzőVezeték"/>
        <xs:field xpath="@NemzetKód"/>
    </xs:key>

    <xs:keyref refer="vezeteknevNemzetkodPK" name="tulajdonosFK">
        <xs:selector xpath="rally/AutóTulajdonos"/>
        <xs:field xpath="@NemzetKód"/>
    </xs:keyref>

    <xs:keyref refer="vezeteknevNemzetkodPK" name="nemzetisegekFK">
        <xs:selector xpath="rally/nemzetisegek"/>
        <xs:field xpath="@NemzetKód"/>
    </xs:keyref>

    <!-- Tulajdonosok Key-->
    <xs:key name="TulajdonosAutoNevPK">
        <xs:selector xpath="AutóTulajdonos"/>
        <xs:field xpath="@Autok"/>
    </xs:key>

    <xs:keyref refer="vezeteknevNemzetkodPK" name="autóTipusokFK">
        <xs:selector xpath="rally/autóTipusok"/>
        <xs:field xpath="@Autok"/>
    </xs:keyref>

    <!-- Auto es orszag kapcsolata -->
    <xs:key name="AutokPKSzakasz">
        <xs:selector xpath="autóTipusok"/>
        <xs:field xpath="@Szakasz"/>
    </xs:key>

    <xs:keyref refer="AutokPKSzakasz" name="országSzakaszNevFK">
        <xs:selector xpath="rally/országok"/>
        <xs:field xpath="@Szakasz"/>
    </xs:keyref>

    <!-- Környezet - Auto kapcsolata -->
    <xs:key name="kornyezetPKAutok">
        <xs:selector xpath="kornyezet"/>
        <xs:field xpath="@Autok"/>
    </xs:key>

    <xs:keyref refer="kornyezetPKAutok" name="autokFK">
        <xs:selector xpath="rally/autóTipusok"/>
        <xs:field xpath="@Autok"/>
    </xs:keyref>
</xs:element>

<!-- Autó Tipus -->

<xs:complexType name="autoTipus">
    <xs:attribute name="Autok" type="markaTipsuok"/></xs:attribute>
    <xs:attribute name="Versenyben" type="versenyTipus"/></xs:attribute>
    <xs:attribute name="Szakasz" type="szakaszTipus"/></xs:attribute>
</xs:complexType>

<!-- Márka típusok -->
<xs:simpleType name="markaTipsuok">
    <xs:restriction base="xs:string">
        <xs:enumeration value="bmw_me_e30"/></xs:enumeration>

```

```

        <xs:enumeration value="ford_fiesta_r5"></xs:enumeration>
        <xs:enumeration value="ford_focus_wrc"></xs:enumeration>
        <xs:enumeration value="skoda_fabia_r5"></xs:enumeration>
        <xs:enumeration value="wv_polo_r5"></xs:enumeration>
    </xs:restriction>
</xs:simpleType>
<!-- Környezet típus -->

<xs:complexType name="kornyezetTipus">
    <xs:attribute name="SérültAutó" type="serulesTipus"></xs:attribute>
    <xs:attribute name="Szakasz" type="szakaszTipus"></xs:attribute>
    <xs:attribute name="Autok" type="markaTipsuok"></xs:attribute>
</xs:complexType>

<!-- End of Környezet Típus -->

<!-- Nemzetiség típusok -->

<xs:complexType name="nemzetisegTipus">
    <xs:sequence>
        <xs:element name="cseh" type="xs:string"></xs:element>
        <xs:element name="francia" type="xs:string"></xs:element>
        <xs:element name="német" type="xs:string"></xs:element>
        <xs:element name="olasz" type="xs:string"></xs:element>
        <xs:element name="magyar" type="xs:string"></xs:element>
    </xs:sequence>
    <xs:attribute name="NemzetKód" type="xs:int"></xs:attribute>
</xs:complexType>

<!-- End of Nemzetiség típusok -->

<!-- Ország típusok -->

<xs:complexType name="országTipus">
    <xs:attribute name="országID" type="xs:int"></xs:attribute>
    <xs:attribute name="Szakasz" type="szakaszTipus"></xs:attribute>
</xs:complexType>

<!-- End of Ország típusok -->

<!-- Szakasz típusok -->

<xs:complexType name="szakaszJellemTipus">
    <xs:attribute name="Szakasz" type="szakaszTipus"></xs:attribute>
    <xs:attribute name="Hőmérséklet"
type="homersekletTipus"></xs:attribute>
    <xs:attribute name="Hossz" type="hosszTipus"></xs:attribute>
    <xs:attribute name="felület" type="feluletTipus"></xs:attribute>
    <xs:attribute name="V kód" type="vkod"></xs:attribute>
</xs:complexType>

<!-- End of Szakasz típusok -->

<!-- Tulajdonos típusok -->

<xs:complexType name="tulajdonTipus">
    <xs:attribute name="Autok" type="markaTipsuok"></xs:attribute>
    <xs:attribute name="NemzetKód" type="xs:int"></xs:attribute>
</xs:complexType>

<!-- End of Tulajdonos típusok -->

```

```

<!-- vezetéknév típusok -->

<xs:complexType name="versenyzoVezetekTipus">
  <xs:sequence>
    <xs:element name="Tanak" type="xs:string"/>
    <xs:element name="Ogier" type="xs:string"/>
    <xs:element name="Rovenpera" type="xs:string"/>
    <xs:element name="Neuville" type="xs:string"/>
    <xs:element name="Perez" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="V kód" type="vkod"/>
  <xs:attribute name="NemzetKód" type="xs:int"/>
</xs:complexType>

<!-- End of vezetéknév típusok -->

<!-- keresztnév típusok -->

<xs:complexType name="versenyzoKeresztTipus">
  <xs:sequence>
    <xs:element name="Ott" type="xs:string"/>
    <xs:element name="Sebastian" type="xs:string"/>
    <xs:element name="Kella" type="xs:string"/>
    <xs:element name="Thierry" type="xs:string"/>
    <xs:element name="Sergio" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="V kód" type="vkod"/>
</xs:complexType>

<!-- End of keresztnév típusok -->

<!-- Versenyben -->
<xs:simpleType name="versenyTipus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="igen"/>
    <xs:enumeration value="nem"/>
  </xs:restriction>
</xs:simpleType>

<!-- Szakasz Típus -->
<xs:simpleType name="szakaszTipus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="francia_szakasz"/>
    <xs:enumeration value="magyar_szakasz"/>
    <xs:enumeration value="nemet_szakasz"/>
    <xs:enumeration value="olasz_szakasz"/>
    <xs:enumeration value="orosz_szakasz"/>
  </xs:restriction>
</xs:simpleType>

<!-- serulesTipus -->

<xs:simpleType name="serulesTipus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="igen"/>
    <xs:enumeration value="nem"/>
  </xs:restriction>
</xs:simpleType>

<!-- homerseklet típusok -->

```

```
<xs:simpleType name="homersekletTipus">
  <xs:restriction base="xs:integer">
    <xs:minInclusive value="-30"></xs:minInclusive>
    <xs:maxInclusive value="45"></xs:maxInclusive>
  </xs:restriction>
</xs:simpleType>

<!-- Hossz típus -->
<xs:simpleType name="hosszTipus">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1"></xs:minInclusive>
    <xs:maxInclusive value="50"></xs:maxInclusive>
  </xs:restriction>
</xs:simpleType>

<!-- Felület típus -->
<xs:simpleType name="feluletTipus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="murva"></xs:enumeration>
    <xs:enumeration value="aszfalt"></xs:enumeration>
    <xs:enumeration value="aszfalt-murva"></xs:enumeration>
  </xs:restriction>
</xs:simpleType>

<!-- Hossz típus -->
<xs:simpleType name="vkod">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1"></xs:minInclusive>
    <xs:maxInclusive value="5"></xs:maxInclusive>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```

2. Feladat:

a) Adatolvasás (DOMReadXUXEJO.java)

- 1. Lépésben Processzéljük az XML Fájlt, amit be szeretnénk olvasni és a támadások elkerülése végett használjuk a

dbf.setFeature(XMLConstants.FEATURE_SECURE_PROCESSING, true); funkciót. Ez alkalmas az XXE támadások kivédésére.

- 2. XML Fájl Parse: .normalize(); funkció segít abban, ha össze vissza vannak sortörések, kihagyások üres karakterek útve, ezeket rendezi át szépen, használata opcionális, én azért használtam, hogy áttekinthető legyen.

3. Végül pedig kiiratjuk a Console-ra a Root (Gyökér) Elementet és lekérdezzük egyesével a Nodeokat (Auto, Szakasz, Ország, Nemzetiség, stb...). Minden új lekérdezésnél //Get <példa> comment sorokat tettem, hogy áttekinthető legyen.

```
package hu.domparsed.XUXEJO;

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

import javax.xml.XMLConstants;
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 DOMReadXUXEJO {
    private static final String FILENAME =
"C:\\java_projects\\DOMParseXUXEJO"
        + "\\src\\hu\\domparsed\\XUXEJO\\1.c_XMLXUXEJO.xml";

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

        try {
            //XML 'Processzélése' biztonságosan
            //Támadások elkerülése mint például: XML External Entities
            (XXE)
            dbf.setFeature(XMLConstants.FEATURE_SECURE_PROCESSING,
true);

            //XML Fájl Parse
            DocumentBuilder db = dbf.newDocumentBuilder();
            Document doc = db.parse(new File(FILENAME));
            //Összevisszaság van a string neveknél a .normalize();
            ezt javítja ki.
            doc.getDocumentElement().normalize();

            System.out.println("Gyökér Element: " +
doc.getDocumentElement().getNodeName());
            System.out.print("-----");

            // get <auto>
```

```

        NodeList auto = doc.getElementsByTagName("auto");

        for (int temp = 0; temp < auto.getLength(); temp++) {
            Node node = auto.item(temp);
            if (node.getNodeType() == Node.ELEMENT_NODE) {
                Element element = (Element) node;
                // Az auto tag attribútuma
                String id = element.getAttribute("id");
                // Szöveg, elemei
                String AutóNév =
element.getElementsByTagName("AutóNév").item(0).getTextContent();
                String Versenyben =
element.getElementsByTagName("Versenyben").item(0).getTextContent();
                String Név =
element.getElementsByTagName("Név").item(0).getTextContent();
                //Kiiratás
                System.out.println("\nJelenlegi element: " +
node.getNodeName());

                System.out.println("Element ID: " + id);
                System.out.println("Autó Neve: " + AutóNév);
                System.out.println("Versenyben van még? " +
Versenyben);

                System.out.println("Szakasz Neve: " + Név);
            }
        }

        // get <kornyezet>
        NodeList környezet =
doc.getElementsByTagName("kornyezet");

        System.out.print("\n-----\n");
        for (int temp = 0; temp < környezet.getLength(); temp++)
        {
            Node node = környezet.item(temp);
            if (node.getNodeType() == Node.ELEMENT_NODE) {
                Element element = (Element) node;
                // Az auto tag attribútuma
                String id = element.getAttribute("id");
                // Szöveg, elemei
                String Sérült_autó =
element.getElementsByTagName("Sérült_autó").item(0).getTextContent();
                String Név =
element.getElementsByTagName("Név").item(0).getTextContent();
                String Autó_Neve =
element.getElementsByTagName("Autó_Neve").item(0).getTextContent();
                //Kiiratás
                System.out.println("\nJelenlegi element: " +
node.getNodeName());

                System.out.println("Element ID: " + id);
                System.out.println("Adott Autó sérült-e: " +
Sérült_autó);

                System.out.println("Szakasz neve: " + Név);
                System.out.println("Autó Neve: " + Autó_Neve);
            }
        }

        // get <nemzetiseg>
        NodeList nemzetiseg =
doc.getElementsByTagName("nemzetiseg");

        System.out.print("\n-----\n");

```

```

        for (int temp = 0; temp < nemzetiseg.getLength(); temp++)
        {
            Node node = nemzetiseg.item(temp);
            if (node.getNodeType() == Node.ELEMENT_NODE) {
                Element element = (Element) node;
                // Az auto tag attribútuma
                String id = element.getAttribute("id");
                // Szöveg, elemei
                String NemzetKód =
element.getElementsByTagName("NemzetKód").item(0).getTextContent();
                String Név =
element.getElementsByTagName("Név").item(0).getTextContent();
                //Kiiratás
                System.out.println("\nJelenlegi element: " +
node.getNodeName());

                System.out.println("Element ID: " + id);
                System.out.println("Nemzetiség kódja: " +
NemzetKód);

                System.out.println("Nemzetiség neve: " + Név);
            }
        }

// get <ország>
NodeList orszag = doc.getElementsByTagName("ország");

System.out.print("\n-----\n");
for (int temp = 0; temp < orszag.getLength(); temp++) {
    Node node = orszag.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        // Az auto tag attribútuma
        String id = element.getAttribute("id");
        // Szöveg, elemei
        String OrszágID =
element.getElementsByTagName("OrszágID").item(0).getTextContent();
        String Név =
element.getElementsByTagName("Név").item(0).getTextContent();
        //Kiiratás
        System.out.println("\nJelenlegi element: " +
node.getNodeName());

        System.out.println("Element ID: " + id);
        System.out.println("Ország kódja: " + OrszágID);
        System.out.println("Országhoz tartozó szakasz
neve: " + Név);
    }
}

// get <szakasz>
NodeList szakasz = doc.getElementsByTagName("szakasz");

System.out.print("\n-----\n");
for (int temp = 0; temp < szakasz.getLength(); temp++) {
    Node node = szakasz.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        // Az auto tag attribútuma
        String id = element.getAttribute("id");
        // Szöveg, elemei
        String Név =
element.getElementsByTagName("Név").item(0).getTextContent();

```



```

        String Hőmérséklet =
element.getElementsByTagName("Hőmérséklet").item(0).getTextContent();
        String Hossz =
element.getElementsByTagName("Hossz").item(0).getTextContent();
        String Felület =
element.getElementsByTagName("Felület").item(0).getTextContent();
        String V kód =
element.getElementsByTagName("V kód").item(0).getTextContent();
        //Kiíratás
        System.out.println("\nJelenlegi element: " +
node.getNodeName());
        System.out.println("Element ID: " + id);
        System.out.println("Szakasz neve: " + Név);
        System.out.println("Szakaszon az adott
hőmérséklet: " + Hőmérséklet + " Celsius");
        System.out.println("Szakasz hossza: " + Hossz + "
KM");
        System.out.println("Szakasz Felülete: " +
Felület);
        System.out.println("Szakasz kódja: " + V kód);
    }
}

// get <tulajdonos>
NodeList tulajdonos =
doc.getElementsByTagName("tulajdonos");

System.out.print("\n-----\n");
for (int temp = 0; temp < tulajdonos.getLength(); temp++)
{
    Node node = tulajdonos.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        // Az auto tag attribútuma
        String id = element.getAttribute("id");
        // Szöveg, elemei
        String AutóNév =
element.getElementsByTagName("AutóNév").item(0).getTextContent();
        String NemzetKód =
element.getElementsByTagName("NemzetKód").item(0).getTextContent();
        //Kiíratás
        System.out.println("\nJelenlegi element: " +
node.getNodeName());
        System.out.println("Element ID: " + id);
        System.out.println("Autó Neve: " + AutóNév);
        System.out.println("Nemzetiség kódja: " +
NemzetKód);
    }
}

// get <versenyző>
NodeList versenyző =
doc.getElementsByTagName("versenyző");

System.out.print("\n-----\n");
for (int temp = 0; temp < versenyző.getLength(); temp++)
{
    Node node = versenyző.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        // Az auto tag attribútuma

```

```

        String id = element.getAttribute("id");
        // Szöveg, elemei
        String Vkód =
element.getElementsByTagName("Vkód").item(0).getTextContent();
        String Vezetéknév =
element.getElementsByTagName("Vezetéknév").item(0).getTextContent();
        String Nemzetkód =
element.getElementsByTagName("Nemzetkód").item(0).getTextContent();
        //Kiiratás
        System.out.println("\nJelenlegi element: " +
node.getNodeName());

        System.out.println("Element ID: " + id);
        System.out.println("Verseny kódja: " + Vkód);
        System.out.println("Vezetéknév: " + Vezetéknév);
        System.out.println("Nemzetiség kódja: " +
Nemzetkód);
    }
}

// get <v_keresztnev>
NodeList keresztnev =
doc.getElementsByTagName("v_keresztnev");

System.out.print("\n-----\n");
for (int temp = 0; temp < keresztnev.getLength(); temp++)
{
    Node node = keresztnev.item(temp);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        // Az auto tag attribútuma
        String id = element.getAttribute("id");
        // Szöveg, elemei
        String Keresztnev =
element.getElementsByTagName("Keresztnev").item(0).getTextContent();
        String Vkód =
element.getElementsByTagName("Vkód").item(0).getTextContent();
        //Kiiratás
        System.out.println("\nJelenlegi element: " +
node.getNodeName());

        System.out.println("Element ID: " + id);
        System.out.println("Verseny kódja: " + Vkód);
        System.out.println("Keresztnev: " + Keresztnev);
    }
}

    } catch (ParserConfigurationException | SAXException |
IOException e) {
        e.printStackTrace();
    }
}
}

```

b) Adatmódosítás (DOMModifyXUXEJO.java)

- 1. Először megadtuk annak Fájlnak az elérési helyét (File inputFile = ...), amivel dolgozni fogunk.
- 2. Miután beolvastuk az adatokat Node cars = doc.getFirstChild(); Funkcióval legérdezzük a legelső Node-ot.
- 3. Következő lépésben módosítjuk a brand="bmw" tipust "Opel"-re.
- 4. Elindítunk egy Loopot az auto Node-on keresztül és az "AutóNév" elementek közül azokat módosítjuk "corsa_r5"-re, amik "bmw_me_e30" néven futottak.
- 5. Ezek után minden 2. ID-jű Elementet törölünk minden Node-ban.
- 6. Végül printeljük a módosított Fájlt.

```
package hu.domparse.XUXEJO;

import java.io.File;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
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;

public class DOMModifyXUXEJO {

    public static void main(String argv[]) {

        try {
            File inputFile = new File("C:\\java_projects\\DOMParseXUXEJO"
                                     + "\\src\\hu\\domparse\\XUXEJO\\1.c_XMLXUXEJO.xml");
            DocumentBuilderFactory docFactory =
DocumentBuilderFactory.newInstance();
            DocumentBuilder docBuilder = docFactory.newDocumentBuilder();
            Document doc = docBuilder.parse(inputFile);

            Node cars = doc.getFirstChild();
            Node auto = doc.getElementsByTagName("auto").item(0);

            // update brand attribute
            NamedNodeMap attr = auto.getAttributes();
            Node nodeAttr = attr.getNamedItem("brand");
            nodeAttr.setTextContent("Opel");

            // loop the auto child node
            NodeList list = auto.getChildNodes();

            for (int temp = 0; temp < list.getLength(); temp++) {
                Node node = list.item(temp);
                if (node.getNodeType() == Node.ELEMENT_NODE) {
                    Element eElement = (Element) node;
```

```

        if ("AutóNév".equals(eElement.getNodeName())) {
            if ("bmw_m3_e30".equals(eElement.getTextContent())) {
                eElement.setTextContent("corsa_r5");
            }
        }
    }
}

NodeList childNodes = cars.getChildNodes();

for(int count = 0; count < childNodes.getLength(); count++) {
    Node node = childNodes.item(count);
    if (node.getNodeType() == Node.ELEMENT_NODE) {
        Element element = (Element) node;
        String id = element.getAttribute("id");

        if ("2".equals(id))
            cars.removeChild(node);
    }
}

// write the content on console
TransformerFactory transformerFactory =
TransformerFactory.newInstance();
Transformer transformer = transformerFactory.newTransformer();
DOMSource source = new DOMSource(doc);
System.out.println("-----Modified File-----");
StreamResult consoleResult = new StreamResult(System.out);
transformer.transform(source, consoleResult);
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```

c) Adatlekérdezés (DOMQueryXUXEJO.java)

- 1. Az Adatmódosítás példához hasonlóan megadjuk egy változóban a fájlnek az elérési útját.
- 2. Előzőekhez hasonlóan .normalize() funkció használata, Root element kiírása.
- 3. Elindítunk egy ciklust az „auto“ node-on, kiírjuk első sorban a jelenlegi elementet, ahol tartunk éppen. Lekérdezzük az „auto“ node „brand“ típusát és kiírjuk a konzolra autobrand-ként. Utána áttérünk az „AutóNév“ elementekre, ahol egy for ciklus segítségével kiírjuk az autó Márkáját és hogy milyen szakaszra van felkészítve az autó .

```
package hu.domparse.XUXEJO;

import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.DocumentBuilder;
import org.w3c.dom.Document;
import org.w3c.dom.NodeList;
import org.w3c.dom.Node;
import org.w3c.dom.Element;
import java.io.File;

public class DOMQueryXUXEJO {

    public static void main(String argv[]) {

        try {

            //XML Fajl Parse
            File inputFile = new File("C:\\java_projects\\DOMParseXUXEJO"
                                     + "\\src\\hu\\domparse\\XUXEJO\\1.c_XMLXUXEJO.xml");
            DocumentBuilderFactory dbFactory =
DocumentBuilderFactory.newInstance();
            DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
            Document doc = dBuilder.parse(inputFile);
            doc.getDocumentElement().normalize();

            //Root Element es azon belül az auto node lekerese
            System.out.print("Root element: ");
            System.out.println(doc.getDocumentElement().getNodeName());
            NodeList auto = doc.getElementsByTagName("auto");
            System.out.println("-----");

            for (int temp = 0; temp < auto.getLength(); temp++) {
                //For ciklus hogy vegig loopoljunk az osszes elementen
                Node nNode = auto.item(temp);
                System.out.println("\nCurrent Element: ");
                System.out.print(nNode.getNodeName());

                if (nNode.getNodeType() == Node.ELEMENT_NODE) {
                    //Egy IF-fel megvizsgaljuk, hogy melyik brandet kerjuk le -
                    Ennek a neve AutóNév
                    Element eElement = (Element) nNode;
                    System.out.print("brand: ");
                    System.out.println(eElement.getAttribute("brand"));
                    NodeList carNameList =
eElement.getElementsByTagName("AutóNév");

                    for (int count = 0; count < carNameList.getLength();
count++) {
```

```

//Egy újabb For ciklus indítása mellyel kiíratjuk a
lekerd adatokat
Node node1 = carNameList.item(count);

if (node1.getNodeType() == node1.ELEMENT_NODE) {
    Element car = (Element) node1;
    System.out.print("Autó Márkája: ");
    System.out.println(car.getTextContent());
    System.out.print("Milyen szakaszra van felkészítve az
autó: ");

    System.out.println(car.getAttribute("type"));
}
}
}
}
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```