

Jegyzőkönyv
Adatkezelés XML környezetben
I. Féléves feladat

Készítette: **Bujnóczki Bence**
Neptunkód: **B00P5Y**

A feladat leírása:

A feladatban egy állatkert adatbázis modelljét hozom létre. Az „Állat” egyed tárolja az állatkert állatait. Minden állatnak egy lakhelye van. A „Név” tulajdonság opcionális, mert van olyan állat aminek nem adnak nevet, vagy még nem adtak nevet Pl.: halak, giliszták, most született állatok. A „Kor” a „Születés ideje”-ből származtatott tulajdonság.

A „Gondozó” egyedben nyilván vannak tartva az állatok gondozói. Ők végzik a gondozást és az élőhelyek takarítását is. Az „SZszám” tulajdonság a dolgozóval kötött szerződés számát jelöli. **Egy gondozó több állatot is** gondozhat és **több élőhelyet is** takaríthat. **Egy állatnak is lehet több gondozója**, és **egy élőhelynek is lehet több takarítója**. Erre azért van szükség, mert ha egy gondozó épp szabadságon van vagy beteg, akkor egy másiknak kell elvégeznie a munkáját.

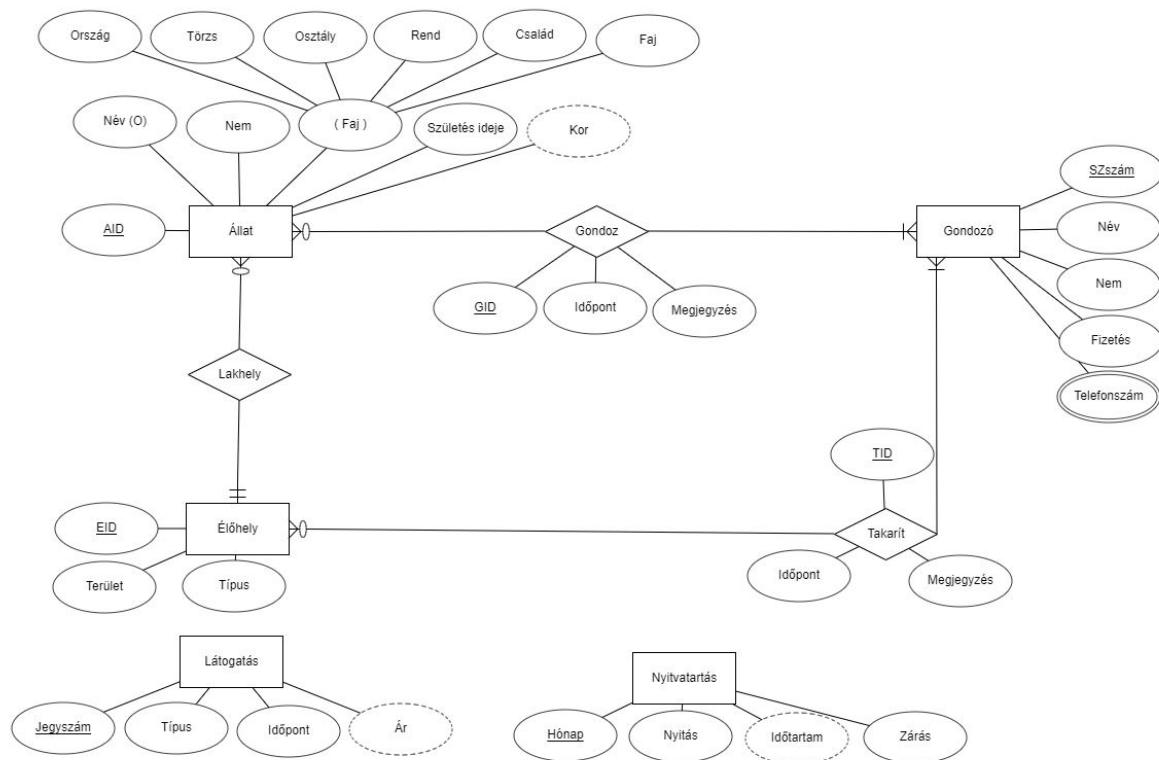
A „Takarít” és „Gondoz” kapcsolatok tartalmazznak elsődleges kulcsot, az egyedi azonosítás miatt.

Az „Élőhely” egyed az állatok lakóhelyeit tárolja. A „Típus” mező tárolja az élőhely fajtáját, ami lehet például erdős, madárketrec, sarkvidéki, stb.

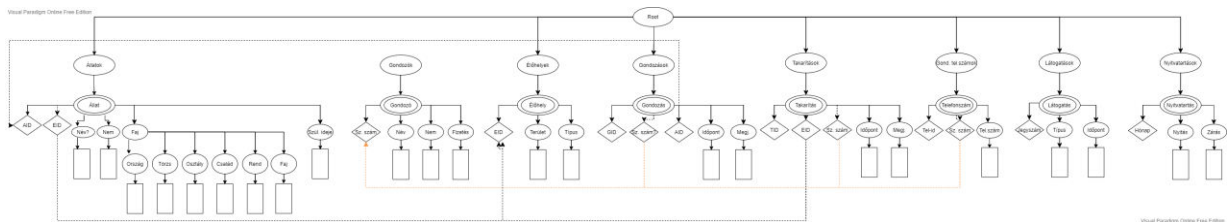
A „Nyitvatartás” egyed az állatkert nyitvatartási idejét tárolja hónapra lebontva. Pl.: Szeptember 8:00-18:00. Az „Időtartam” egy számított érték. Kiszámítása: „Zárás” – „Nyitás”.

A „Látogatás” egyed a vásárolt jegyeket tartja nyilván. Az elsődleges kulcsa a „Jegyszám” tulajdonság, ami a jegy sorszáma, és mindig eggyel nő. Az „Időpont” a vásárlás idejét jelöli. Az „Ár” egy származtatott tulajdonság, ami a jegy típusából következik.

Az adatbázis ER modellje:



Az adatbázis konvertálása XDM modellre:



Az XDM modell alapján XML dokumentum készítése:

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

```
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="XMLSchemaB00P5Y.xsd">
```

```
<allatok>
```

```
<allat aid="a-1" eid="e-5">
```

```
<nev>Pihe</nev>
```

```
<nem>Hím</nem>
```

```
<faj>
```

```
<ország>Állatok</ország>
```

```

    <torzs>Gerinchúrosok</torzs>

    <osztaly>Emlősök</osztaly>

    <rend>Ragadozók</rend>

    <csalad>Macskafélék</csalad>

    <faj>Vadmacska</faj>

</faj>

<szuletes-ideje>

2021-04-20

</szuletes-ideje>

</allat>

<allat aid="a-2" eid="e-5">

    <nev>Cirmi</nev>

    <nem>Nőstény</nem>

    <faj>

        <ország>Állatok</ország>

        <torzs>Gerinchúrosok</torzs>

        <osztaly>Emlősök</osztaly>

        <rend>Ragadozók</rend>

        <csalad>Macskafélék</csalad>

        <faj>Vadmacska</faj>

    </faj>

    <szuletes-ideje>

2021-08-19

</szuletes-ideje>

</allat>

<allat aid="a-3" eid="e-3">

    <nev>Füles</nev>

    <nem>Hím</nem>

    <faj>

        <ország>Állatok</ország>

        <torzs>Gerinchúrosok</torzs>

        <osztaly>Emlősök</osztaly>

        <rend>Ragadozók</rend>

        <csalad>Kutyafélék</csalad>

        <faj>Farkas</faj>

```

```

</faj>

<szuletes-ideje>

2016-05-11

</szuletes-ideje>

</allat>

<allat aid="a-4" eid="e-3">

  <nev>Vadóc</nev>

  <nem>Nőstény</nem>

  <faj>

    <ország>Állatok</ország>

    <torzs>Gerinchúrosok</torzs>

    <osztaly>Emlősök</osztaly>

    <rend>Ragadozók</rend>

    <csalad>Kutyafélék</csalad>

    <faj>Farkas</faj>

  </faj>

  <szuletes-ideje>

  2017-08-30

  </szuletes-ideje>

</allat>

<allat aid="a-5" eid="e-3">

  <nev>Vadóc</nev>

  <nem>Nőstény</nem>

  <faj>

    <ország>Állatok</ország>

    <torzs>Gerinchúrosok</torzs>

    <osztaly>Emlősök</osztaly>

    <rend>Ragadozók</rend>

    <csalad>Kutyafélék</csalad>

    <faj>Farkas</faj>

  </faj>

  <szuletes-ideje>

  2017-08-30

  </szuletes-ideje>

</allat>

```

```

<allat aid="a-6" eid="e-2">
  <nev>Béla</nev>
  <nem>Hím</nem>
  <faj>
    <ország>Állatok</ország>
    <torzs>Gerinchúrosok</torzs>
    <osztaly>Emlősök</osztaly>
    <rend>Párosujjú patások</rend>
    <csalad>Disznófélék</csalad>
    <faj>Vaddisznó</faj>
  </faj>
  <szuletes-ideje>
    2016-10-26
  </szuletes-ideje>
</allat>

<allat aid="a-7" eid="e-2">
  <nev>Barnabás</nev>
  <nem>Hím</nem>
  <faj>
    <ország>Állatok</ország>
    <torzs>Gerinchúrosok</torzs>
    <osztaly>Emlősök</osztaly>
    <rend>Párosujjú patások</rend>
    <csalad>Disznófélék</csalad>
    <faj>Vaddisznó</faj>
  </faj>
  <szuletes-ideje>2016-10-26</szuletes-ideje>
</allat>

<allat aid="a-8" eid="e-2">
  <nev>Cecília</nev>
  <nem>Nőstény</nem>
  <faj>
    <ország>Állatok</ország>
    <torzs>Gerinchúrosok</torzs>
    <osztaly>Emlősök</osztaly>

```

```

    <rend>Párosujjú patások</rend>

    <csalad>Disznófélék</csalad>

    <faj>Vaddisznó</faj>

</faj>

    <szuletes-ideje>2015-12-03</szuletes-ideje>

</allat>

<allat aid="a-9" eid="e-2">

    <nev>Jessica</nev>

    <nem>Nőstény</nem>

    <faj>

        <ország>Állatok</ország>

        <torzs>Gerinchúrosok</torzs>

        <osztaly>Emlősök</osztaly>

        <rend>Párosujjú patások</rend>

        <csalad>Disznófélék</csalad>

        <faj>Vaddisznó</faj>

    </faj>

    <szuletes-ideje>

        2018-01-26

    </szuletes-ideje>

</allat>

<allat aid="a-10" eid="e-1">

    <nev>Totyi</nev>

    <nem>Hím</nem>

    <faj>

        <ország>Állatok</ország>

        <torzs>Gerinchúrosok</torzs>

        <osztaly>Madarak</osztaly>

        <rend>Pingvinalakúak</rend>

        <csalad>Pingvinfélék</csalad>

        <faj>Császárpingvin</faj>

    </faj>

    <szuletes-ideje>

        2011-05-11

    </szuletes-ideje>

```

```

</allat>
<allat aid="a-11" eid="e-1">
  <nev>Tipegő</nev>
  <nem>Hím</nem>
  <faj>
    <ország>Állatok</ország>
    <torzs>Gerinchúrosok</torzs>
    <osztaly>Madarak</osztaly>
    <rend>Pingvinalakúak</rend>
    <csalad>Pingvinfélék</csalad>
    <faj>Császárpingvin</faj>
  </faj>
  <szuletes-ideje>
    2013-07-11
  </szuletes-ideje>
</allat>
<allat aid="a-12" eid="e-4">
  <nev>Zsolti</nev>
  <nem>Hím</nem>
  <faj>
    <ország>Állatok</ország>
    <torzs>Gerinchúrosok</torzs>
    <osztaly>Madarak</osztaly>
    <rend>Pingvinalakúak</rend>
    <csalad>Úszólábúak</csalad>
    <faj>Fóka</faj>
  </faj>
  <szuletes-ideje>
    2015-05-06
  </szuletes-ideje>
</allat>
<allat aid="a-13" eid="e-4">
  <nev>Szandi</nev>
  <nem>Nőstény</nem>
  <faj>

```



```

<ország>Állatok</ország>

<torzs>Gerinchúrosok</torzs>

<osztaly>Madarak</osztaly>

<rend>Pingvinalakúak</rend>

<csalad>Úszólábúak</csalad>

<faj>Fóka</faj>

</faj>

<szuletes-ideje>

2014-12-05

</szuletes-ideje>

</allat>

<allat aid="a-14" eid="e-6">

<nev>Fütyülős</nev>

<nem>Hím</nem>

<faj>

<ország>Állatok</ország>

<torzs>Gerinchúrosok</torzs>

<osztaly>Madarak</osztaly>

<rend>Lúdalakúak</rend>

<csalad>Récefélék</csalad>

<faj>Tőkés réce</faj>

</faj>

<szuletes-ideje>

2018-03-11

</szuletes-ideje>

</allat>

<allat aid="a-15" eid="e-1">

<nem>Nőstény</nem>

<faj>

<ország>Állatok</ország>

<torzs>Gerinchúrosok</torzs>

<osztaly>Madarak</osztaly>

<rend>Pingvinalakúak</rend>

<csalad>Pingvinfélék</csalad>

<faj>Császárpingvin</faj>

```

```

</faj>

<szuletes-ideje>

2019-11-15

</szuletes-ideje>

</allat>

<allat aid="a-16" eid="e-6">

  <nem>Hím</nem>

  <faj>

    <ország>Állatok</ország>

    <torzs>Gerinchúrosok</torzs>

    <osztaly>Madarak</osztaly>

    <rend>Lúdalakúak</rend>

    <csalad>Récefélék</csalad>

    <faj>Tőkés réce</faj>

  </faj>

  <szuletes-ideje>

2019-03-14

</szuletes-ideje>

</allat>

<allat aid="a-17" eid="e-6">

  <nem>Nőstény</nem>

  <faj>

    <ország>Állatok</ország>

    <torzs>Gerinchúrosok</torzs>

    <osztaly>Madarak</osztaly>

    <rend>Lúdalakúak</rend>

    <csalad>Récefélék</csalad>

    <faj>Tőkés réce</faj>

  </faj>

  <szuletes-ideje>

2019-05-31

</szuletes-ideje>

</allat>

<allat aid="a-18" eid="e-6">

  <nem>Nőstény</nem>

```

```

    <faj>
      <ország>Állatok</ország>
      <torzs>Gerinchúrosok</torzs>
      <osztaly>Madarak</osztaly>
      <rend>Lúdalakúak</rend>
      <csalad>Récefélék</csalad>
      <faj>Tőkés réce</faj>
    </faj>
    <szuletes-ideje>
      2017-03-11
    </szuletes-ideje>
  </allat>
  <allat aid="a-19" eid="e-6">
    <nem>Nőstény</nem>
    <faj>
      <ország>Állatok</ország>
      <torzs>Gerinchúrosok</torzs>
      <osztaly>Madarak</osztaly>
      <rend>Lúdalakúak</rend>
      <csalad>Récefélék</csalad>
      <faj>Tőkés réce</faj>
    </faj>
    <szuletes-ideje>
      2016-03-20
    </szuletes-ideje>
  </allat>
</allatok>
<gondozok>
  <gondozo személyi-szam="123456AA">
    <nev>Dolgos Jóska</nev>
    <nem>Férfi</nem>
    <fizetes>150000</fizetes>
  </gondozo>
  <gondozo személyi-szam="123456BA">
    <nev>Radó Fülöp</nev>

```

```
<nem>Férfi</nem>

<fizetes>140000</fizetes>

</gondozo>

<gondozo személyi-szam="654321BA">

  <nev>Nyers Gabriella</nev>

  <nem>Nő</nem>

  <fizetes>140000</fizetes>

</gondozo>

<gondozo személyi-szam="654323BA">

  <nev>Süveges Krisztina</nev>

  <nem>Nő</nem>

  <fizetes>160000</fizetes>

</gondozo>

<gondozo személyi-szam="654323CC">

  <nev>Levél Gábor</nev>

  <nem>Férfi</nem>

  <fizetes>180000</fizetes>

</gondozo>

</gondozok>

<elohelyek>

<elohely eid="e-1">

  <terulet>13</terulet>

  <tipus>Sarkvidéki</tipus>

</elohely>

<elohely eid="e-2">

  <terulet>30</terulet>

  <tipus>Erdős</tipus>

</elohely>

<elohely eid="e-3">

  <terulet>32</terulet>

  <tipus>Erdős</tipus>

</elohely>

<elohely eid="e-4">

  <terulet>20</terulet>

  <tipus>Sarkvidéki</tipus>
```

```

</elhely>

<elhely eid="e-5">

  <terulet>10</terulet>

  <tipus>Erdős</tipus>

</elhely>

<elhely eid="e-6">

  <terulet>8</terulet>

  <tipus>Madárketrec</tipus>

</elhely>

</elhelyek>

<gondozasok>

  <gondozas gid="g-1" személyi-szam="123456BA" aid="a-19">

    <idopont>2019-11-10T08:52:00</idopont>

    <megjegyzes>Hamar megvolt</megjegyzes>

  </gondozas>

  <gondozas gid="g-2" személyi-szam="123456BA" aid="a-10">

    <idopont>2019-11-10T08:59:00</idopont>

  </gondozas>

  <gondozas gid="g-3" személyi-szam="654321BA" aid="a-7">

    <idopont>2019-11-10T08:53:00</idopont>

  </gondozas>

  <gondozas gid="g-4" személyi-szam="654321BA" aid="a-8">

    <idopont>2019-11-10T08:58:00</idopont>

  </gondozas>

  <gondozas gid="g-5" személyi-szam="654323BA" aid="a-15">

    <idopont>2019-11-10T09:02:00</idopont>

  </gondozas>

  <gondozas gid="g-6" személyi-szam="654323BA" aid="a-5">

    <idopont>2019-11-10T09:05:00</idopont>

  </gondozas>

  <gondozas gid="g-7" személyi-szam="654323BA" aid="a-6">

    <idopont>2019-11-10T09:05:00</idopont>

    <megjegyzes>Ideges volt az állat</megjegyzes>

  </gondozas>

  <gondozas gid="g-8" személyi-szam="654323BA" aid="a-1">

```

```
<idopont>2019-11-10T09:10:00</idopont>

</gondozas>

<gondozas gid="g-9" személyi-szam="654323CC" aid="a-2">

  <idopont>2019-11-10T09:15:00</idopont>

</gondozas>

<gondozas gid="g-10" személyi-szam="654323CC" aid="a-13">

  <idopont>2019-11-10T08:31:00</idopont>

</gondozas>

<gondozas gid="g-11" személyi-szam="123456BA" aid="a-3">

  <idopont>2019-11-10T08:36:00</idopont>

</gondozas>

<gondozas gid="g-12" személyi-szam="123456BA" aid="a-4">

  <idopont>2019-11-10T08:39:00</idopont>

</gondozas>

</gondozasok>

<takaritasok>

  <takaritas tid="t-1" eid="e-1" személyi-szam="123456BA">

    <idopont>2019-11-10T09:53:00</idopont>

  </takaritas>

  <takaritas tid="t-2" eid="e-6" személyi-szam="654323BA">

    <idopont>2019-11-10T11:05:00</idopont>

    < megjegyzes>Eltört a felmosó</ megjegyzes>

  </takaritas>

  <takaritas tid="t-3" eid="e-4" személyi-szam="654323CC">

    <idopont>2019-11-10T10:58:00</idopont>

  </takaritas>

  <takaritas tid="t-4" eid="e-2" személyi-szam="654323CC">

    <idopont>2019-11-10T10:09:00</idopont>

  </takaritas>

  <takaritas tid="t-5" eid="e-3" személyi-szam="654323CC">

    <idopont>2019-11-11T10:00:00</idopont>

  </takaritas>

</takaritasok>

<gondozok-telefonszamok>

  < telefonszam telid="1" személyi-szam="123456BA">
```

```

    <telefonszam>06202825457</telefonszam>
  </telefonszam>
  <telefonszam telid="2" személyi-szam="123456AA">
    <telefonszam>+36201234567</telefonszam>
  </telefonszam>
  <telefonszam telid="3" személyi-szam="654321BA">
    <telefonszam>06705555555</telefonszam>
  </telefonszam>
  <telefonszam telid="4" személyi-szam="654323BA">
    <telefonszam>06301880254</telefonszam>
  </telefonszam>
  <telefonszam telid="5" személyi-szam="654323CC">
    <telefonszam>06705750215</telefonszam>
  </telefonszam>
  <telefonszam telid="6" személyi-szam="654323CC">
    <telefonszam>+36207550760</telefonszam>
  </telefonszam>
</gondozok-telefonszamok>
<latogatasok>
  <latogatas jegyszam="j-1">
    <tipus>Normál</tipus>
    <idopont>2019.11.10. 12:51</idopont>
  </latogatas>
  <latogatas jegyszam="j-2">
    <tipus>Normál</tipus>
    <idopont>2019.11.10. 14:50</idopont>
  </latogatas>
  <latogatas jegyszam="j-3">
    <tipus>Nyugdíjas</tipus>
    <idopont>2019.11.11. 11:11</idopont>
  </latogatas>
  <latogatas jegyszam="j-4">
    <tipus>Diák</tipus>
    <idopont>2019.11.11. 13:01</idopont>
  </latogatas>

```

```
<latogatas jegyszam="j-5">
  <tipus>Normál</tipus>
  <idopont>2019.11.12. 10:51</idopont>
</latogatas>

<latogatas jegyszam="j-6">
  <tipus>Normál</tipus>
  <idopont>2019.11.12. 10:21</idopont>
</latogatas>
</latogatasok>

<nyitvatartasok>
  <nyitvatartas honap="1">
    <nyitas>10:00:00</nyitas>
    <zaras>17:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="2">
    <nyitas>10:00:00</nyitas>
    <zaras>17:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="3">
    <nyitas>09:00:00</nyitas>
    <zaras>17:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="4">
    <nyitas>09:00:00</nyitas>
    <zaras>18:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="5">
    <nyitas>08:00:00</nyitas>
    <zaras>18:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="6">
    <nyitas>08:00:00</nyitas>
    <zaras>18:00:00</zaras>
  </nyitvatartas>
  <nyitvatartas honap="7">
```



```

    <nyitas>08:00:00</nyitas>

    <zaras>19:00:00</zaras>

</nyitvatartas>

<nyitvatartas honap="8">

    <nyitas>08:00:00</nyitas>

    <zaras>19:00:00</zaras>

</nyitvatartas>

<nyitvatartas honap="9">

    <nyitas>08:00:00</nyitas>

    <zaras>18:00:00</zaras>

</nyitvatartas>

<nyitvatartas honap="10">

    <nyitas>09:00:00</nyitas>

    <zaras>18:00:00</zaras>

</nyitvatartas>

<nyitvatartas honap="11">

    <nyitas>09:00:00</nyitas>

    <zaras>17:00:00</zaras>

</nyitvatartas>

<nyitvatartas honap="12">

    <nyitas>10:00:00</nyitas>

    <zaras>17:00:00</zaras>

</nyitvatartas>

</nyitvatartasok>

</root>

```

Az XML dokumentum alapján XMLSchema készítése:

```

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="root">

    <xs:complexType>

      <xs:sequence>

        <xs:element name="allatok">

          <xs:complexType>

            <xs:sequence>

              <xs:element name="allat" minOccurs="0" maxOccurs="unbounded">

```

```

<xs:complexType>

  <xs:sequence>

    <xs:element name="nev" type="xs:string" minOccurs="0" maxOccurs="1"></xs:element>

    <xs:element name="nem" type="allatNem"></xs:element>

    <xs:element name="faj">

      <xs:complexType>

        <xs:sequence>

          <xs:element name="orszag" type="xs:string"></xs:element>

          <xs:element name="torzs" type="xs:string"></xs:element>

          <xs:element name="osztaly" type="xs:string"></xs:element>

          <xs:element name="rend" type="xs:string"></xs:element>

          <xs:element name="csalad" type="xs:string"></xs:element>

          <xs:element name="faj" type="xs:string"></xs:element>

        </xs:sequence>

      </xs:complexType>

    </xs:element>

    <xs:element name="szuletes-ideje" type="xs:date"></xs:element>

  </xs:sequence>

  <xs:attribute name="aid" type="xs:string" use="required"></xs:attribute>

  <xs:attribute name="eid" type="xs:string" use="required"></xs:attribute>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="gondozok">

  <xs:complexType>

    <xs:sequence>

      <xs:element name="gondozo" minOccurs="0" maxOccurs="unbounded">

        <xs:complexType>

          <xs:sequence>

            <xs:element name="nev" type="xs:string"></xs:element>

            <xs:element name="nem" type="nem"></xs:element>

            <xs:element name="fizetes" type="xs:int"></xs:element>

          </xs:sequence>

        </xs:complexType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

</xs:element>

```

```

    <xs:attribute name="szemelyi-szam" type="szemelyiSzam" use="required"></xs:attribute>

  </xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="elohelyek">

  <xs:complexType>

    <xs:sequence>

      <xs:element name="elohely" minOccurs="0" maxOccurs="unbounded">

        <xs:complexType>

          <xs:sequence>

            <xs:element name="terulet" type="xs:float"></xs:element>

            <xs:element name="tipus" type="elohelyTipus"></xs:element>

          </xs:sequence>

          <xs:attribute name="eid" type="xs:string" use="required"></xs:attribute>

        </xs:complexType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

</xs:element>

<xs:element name="gondozasok">

  <xs:complexType>

    <xs:sequence>

      <xs:element name="gondozas" minOccurs="0" maxOccurs="unbounded">

        <xs:complexType>

          <xs:sequence>

            <xs:element name="idopont" type="xs:dateTime"></xs:element>

            <xs:element name="megjegyzes" type="xs:string" minOccurs="0" maxOccurs="1"></xs:element>

          </xs:sequence>

          <xs:attribute name="gid" type="xs:string" use="required"></xs:attribute>

          <xs:attribute name="szemelyi-szam" type="szemelyiSzam" use="optional"></xs:attribute>

          <xs:attribute name="aid" type="xs:string" use="required"></xs:attribute>

        </xs:complexType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

</xs:element>

```

```

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="takaritasok">

  <xs:complexType>

    <xs:sequence>

      <xs:element name="takaritas" minOccurs="0" maxOccurs="unbounded">

        <xs:complexType>

          <xs:sequence>

            <xs:element name="idopont" type="xs:dateTime"></xs:element>

            <xs:element name="megjegyzes" type="xs:string" minOccurs="0" maxOccurs="1"></xs:element>

          </xs:sequence>

          <xs:attribute name="tid" type="xs:string" use="required"></xs:attribute>

          <xs:attribute name="eid" type="xs:string" use="required"></xs:attribute>

          <xs:attribute name="szemelyi-szam" type="szemelyiSzam" use="required"></xs:attribute>

        </xs:complexType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

</xs:element>

<xs:element name="gondozok-telefonszamok">

  <xs:complexType>

    <xs:sequence>

      <xs:element name="telefonszam" minOccurs="0" maxOccurs="unbounded">

        <xs:complexType>

          <xs:sequence>

            <xs:element name="telefonszam" type="hungarianPhoneNumber"></xs:element>

          </xs:sequence>

          <xs:attribute name="telid" type="xs:string" use="required"></xs:attribute>

          <xs:attribute name="szemelyi-szam" type="szemelyiSzam" use="required"></xs:attribute>

        </xs:complexType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

</xs:element>

```

```

<xs:element name="latogatasok">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="latogatas" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="tipus"></xs:element>
            <xs:element name="idopont"></xs:element>
          </xs:sequence>
          <xs:attribute name="jegyszam" type="xs:string"></xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="nyitvatartasok">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="nyitvatartas" minOccurs="12" maxOccurs="12">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="nyitas" type="xs:time"></xs:element>
            <xs:element name="zaras" type="xs:time"></xs:element>
          </xs:sequence>
          <xs:attribute name="honap" type="honap" use="required"></xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:sequence>
  </xs:sequence>
</xs:sequence>
</xs:complexType>

<xs:key name="key_allat">
  <xs:selector xpath="allatok/allat"></xs:selector>

```

```

<xs:field xpath="@aid"></xs:field>

</xs:key>

<xs:key name="key_gondozo">

<xs:selector xpath="gondozok/gondozo"></xs:selector>

<xs:field xpath="@szemelyi-szam"></xs:field>

</xs:key>

<xs:key name="key_elohely">

<xs:selector xpath="elohelyek/elohely"></xs:selector>

<xs:field xpath="@eid"></xs:field>

</xs:key>

<xs:key name="key_gondozas">

<xs:selector xpath="gondozasok/gondozas"></xs:selector>

<xs:field xpath="@gid"></xs:field>

</xs:key>

<xs:key name="key_takaritas">

<xs:selector xpath="takaritasok/takaritas"></xs:selector>

<xs:field xpath="@tid"></xs:field>

</xs:key>

<xs:key name="key_gondozo-telefonszam">

<xs:selector xpath="gondozok-telefonszamok/telefonszam"></xs:selector>

<xs:field xpath="@telid"></xs:field>

</xs:key>

<xs:key name="key_latogatas">

<xs:selector xpath="latogatasok/latogatas"></xs:selector>

<xs:field xpath="@jegyszam"></xs:field>

</xs:key>

<xs:key name="key_nyitvatartas">

<xs:selector xpath="nyitvatartasok/nyitvatartas"></xs:selector>

<xs:field xpath="@honap"></xs:field>

</xs:key>

<xs:keyref name="FK_allat_elohely" refer="key_elohely">

<xs:selector xpath="allatok/allat"></xs:selector>

<xs:field xpath="@eid"></xs:field>

</xs:keyref>

```

```

<xs:keyref name="FK_gondozas_gondozo" refer="key_gondozo">

  <xs:selector xpath="gondozasok/gondozas"></xs:selector>

  <xs:field xpath="@szemelyi-szam"></xs:field>

</xs:keyref>

<xs:keyref name="FK_gondozas_allat" refer="key_allat">

  <xs:selector xpath="gondozasok/gondozas"></xs:selector>

  <xs:field xpath="@aid"></xs:field>

</xs:keyref>

<xs:keyref name="FK_takaritas_elohely" refer="key_elohely">

  <xs:selector xpath="takaritasok/takaritas"></xs:selector>

  <xs:field xpath="@eid"></xs:field>

</xs:keyref>

<xs:keyref name="FK_takaritas_gondozo" refer="key_gondozo">

  <xs:selector xpath="takaritasok/takaritas"></xs:selector>

  <xs:field xpath="@szemelyi-szam"></xs:field>

</xs:keyref>

<xs:keyref name="FK_gondozok-telefonszamok_gondozo" refer="key_gondozo">

  <xs:selector xpath="gondozok-telefonszamok/telefonszam"></xs:selector>

  <xs:field xpath="@szemelyi-szam"></xs:field>

</xs:keyref>

</xs:element>

<xs:simpleType name="hungarianPhoneNumber">

  <xs:restriction base="xs:string">

    <xs:pattern value="(06){1}(20|30|70){1}[0-9]{7}"></xs:pattern>

    <xs:pattern value="(\+36){1}(20|30|70){1}[0-9]{7}"></xs:pattern>

  </xs:restriction>

</xs:simpleType>

<xs:simpleType name="elohelyTipus">

  <xs:restriction base="xs:string">

    <xs:pattern value="Sarkvidéki"></xs:pattern>

    <xs:pattern value="Erdős"></xs:pattern>

    <xs:pattern value="Madárketrec"></xs:pattern>

  </xs:restriction>

</xs:simpleType>

<xs:simpleType name="nem">

```

```

<xs:restriction base="xs:string">
  <xs:pattern value="Nő"></xs:pattern>
  <xs:pattern value="Férfi"></xs:pattern>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="allatNem">
  <xs:restriction base="xs:string">
    <xs:pattern value="Hím"></xs:pattern>
    <xs:pattern value="Nőstény"></xs:pattern>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="szemelyiSzam">
  <xs:restriction base="xs:string">
    <xs:pattern value="[A-Z]{2}[0-9]{6}"></xs:pattern>
    <xs:pattern value="[0-9]{6}[A-Z]{2}"></xs:pattern>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="honap">
  <xs:restriction base="xs:int">
    <xs:minInclusive value="1"></xs:minInclusive>
    <xs:maxInclusive value="12"></xs:maxInclusive>
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

Adatolvasás:

```
package XMLTaskB00P5Y.DOMParseB00P5Y.hu.domparse.B00P5Y;
```

```

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.NamedNodeMap;
import org.w3c.dom.Node;

```



```

import org.w3c.dom.NodeList;

import org.xml.sax.SAXException;

public class DOMReadB00P5Y {

    public static void main(String arg[]) throws SAXException, IOException, ParserConfigurationException {

        // Get file

        File xmlFile = new File("XMLTaskB00P5Y/XMLB00P5Y.xml");

        // Get DocumentBuilder

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();

        DocumentBuilder dBuilder = factory.newDocumentBuilder();

        // Parse document and normalize

        Document doc = dBuilder.parse(xmlFile);

        doc.getDocumentElement().normalize();

        // Call printNodeRecurse on root element

        printNodeRecurse(doc.getDocumentElement());

    }

    /**
     * Overload for printNodeResource,
     * Sets depth = 0
     *
     * @param n
     */
    public static void printNodeRecurse(Node n) {

        printNodeRecurse(n, 0);

    }

    /**
     * Prints node name, attributes, textContent, then calls printNodeRecurse on

```

```

* ELEMENT_NODE children
*
* @param n - node
* @param depth - recursive depth
*/

public static void printNodeRecurse(Node n, int depth) {
    // Determine node name, attributes and child nodes
    String nodeName = n.getNodeName();
    NamedNodeMap nodeAttributes = n.getAttributes();
    NodeList childNodes = n.getChildNodes();

    indent(depth);

    System.out.print("<" + nodeName);

    // Print attributes
    for (int i = 0; i < nodeAttributes.getLength(); i++) {
        Node attribute = nodeAttributes.item(i);
        System.out.print(" " + nodeAttributeToString(attribute));
    }

    System.out.println(">");

    // Iterate on childNodes
    for (int i = 0; i < childNodes.getLength(); i++) {
        Node child = childNodes.item(i);
        short childNodeType = child.getNodeType();

        if (childNodeType != Node.ELEMENT_NODE) {
            if (childNodeType == Node.TEXT_NODE) {
                // If textnode => trim (to avoid empty lines), then print
                String textContent = child.getTextContent().trim();

                if (textContent.length() != 0) {
                    indent(depth + 1);
                    System.out.println(textContent);
                }
            }
            continue;
        }
    }
}

```

```

    }

    // Recursive call
    printNodeRecurse(child, depth + 1);
}

indent(depth);

System.out.println("<\\\" + nodeName + ">");
}

/**
 * Reads all nodename and textcontent and returns it as a formatted String
 *
 * @param attribute - Node - type: Node.ATTRIBUTE_NODE
 * @return
 */
private static String nodeAttributeToString(Node attribute) {
    return attribute.getNodeName() + "=\"" + attribute.getTextContent() + "\"";
}

/**
 * Prints times * tab on the stdout
 *
 * @param times
 */
private static void indent(int times) {
    for (int i = 0; i <= times; i++) {
        System.out.print(" ");
    }
}
}

```

Adatmódosítás:

```
package XMLTaskB00P5Y.DOMParseB00P5Y.hu.domparse.B00P5Y;
```

```

import java.io.File;

import java.io.IOException;

import java.io.UnsupportedEncodingException;


import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerConfigurationException;
import javax.xml.transform.TransformerException;
import javax.xml.transform.TransformerFactory;
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 org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;


enum ElohelyTipus {
    SARKVIDEKI,
    ERDOS,
    MADARKETREC
}

```

```

public class DOMModifyB00P5Y {

    public static void main(String arg[]) throws IOException, ParserConfigurationException, SAXException,
        TransformerConfigurationException, TransformerException, XPathExpressionException {

        // Get input and output file

        File inputXMLFile = new File("XMLTaskB00P5Y/XMLB00P5Y.xml");

        File outputXMLFile = new File("XMLTaskB00P5Y/XMLB00P5Y.out.xml");

        // Get documentBuilder

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();

        DocumentBuilder dBuilder = factory.newDocumentBuilder();

        // Get document and normalize root element

        Document inputDoc = dBuilder.parse(inputXMLFile);

        inputDoc.getDocumentElement().normalize();

        // Apply modifications

        changeAllatNevById(inputDoc, "a-1", "PiheCica");

        addMegjegyzesToGondozasById(inputDoc, "g-11", "Nagyon rossz volt az állat!!");

        addElohely(inputDoc, (float) 25.2, ElohelyTipus.ERDOS);

        // Write to file and console

        write(inputDoc, outputXMLFile);
    }

    /**
     * Write a Document to file and console
     *
     * @param doc
     * @param outputFile
     * @throws TransformerException
     * @throws UnsupportedEncodingException
     */

    private static void write(Document doc, File outputFile)

        throws TransformerException, UnsupportedEncodingException {

```

```

TransformerFactory transformerFactory = TransformerFactory.newInstance();

Transformer transf = transformerFactory.newTransformer();

transf.setOutputProperty(OutputKeys.ENCODING, "UTF-8");

transf.setOutputProperty(OutputKeys.INDENT, "yes");

transf.setOutputProperty("{http://xml.apache.org/xslt}indent-amunt", "2");


DOMSource source = new DOMSource(doc);


StreamResult console = new StreamResult(System.out);

StreamResult file = new StreamResult(outputFile);


transf.transform(source, console);

transf.transform(source, file);
}

/**
 * Állat nevének megváltoztatása id alapján
 *
 * @param doc
 * @param id
 * @param newVal
 */
private static void changeAllatNevById(Document doc, String id, String newVal) throws XPathExpressionException {

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

    String expression = "//allat[@aid='" + id + "']/nev";

    NodeList nodeList = (NodeList) xPath.compile(expression).evaluate(doc, XPathConstants.NODESET);

    if (nodeList.getLength() != 1)

        return;

    Node node = nodeList.item(0);

    node.setTextContent(newVal);
}

```

```

/**
 * Megjegyzés hozzáadása egy gondozáshoz
 *
 * @param doc
 * @param id
 * @param newVal
 */
private static void addMegjegyzesToGondozasById(Document doc, String id, String newVal)
    throws XPathExpressionException {
    XPath xPath = XPathFactory.newInstance().newXPath();
    String expression = "//gondozas[@gid='" + id + "']";
    NodeList nodeList = (NodeList) xPath.compile(expression).evaluate(doc, XPathConstants.NODESET);
    if (nodeList.getLength() != 1)
        return;

    Node node = nodeList.item(0);
    Node megjegyzes = doc.createElement("megjegyzes");
    megjegyzes.setTextContent(newVal);

    node.appendChild(mejjegyzes);
}

/**
 * Élőhely hozzáadása
 *
 * @param newVal
 */
private static void addElohely(Document doc, Float terület, ElohelyTipus tipus) throws XPathExpressionException {
    XPath xPath = XPathFactory.newInstance().newXPath();
    String expression = "//elohelyek";
    String elohelyExpression = "//elohelyek/elohely";
    NodeList nodeList = (NodeList) xPath.compile(expression).evaluate(doc, XPathConstants.NODESET);

```

```

if (nodeList.getLength() != 1)

    return;

NodeList elohelyList = (NodeList) xpath.compile(elohelyExpression).evaluate(doc, XPathConstants.NODESET);
int elohelyCount = elohelyList.getLength();

Node elohelyek = nodeList.item(0);

Node elohelyNode = doc.createElement("elohely");

Node teruletNode = doc.createElement("terulet");

Node tipusNode = doc.createElement("tipus");

// Generate id: elohely.getLength() + 1
((Element) elohelyNode).setAttribute("eid", "e-" + elohelyCount + 1);

String tipusString = "";

teruletNode.setTextContent(terulet.toString());

switch (tipus) {

    case ERDOS:

        tipusString = "Erdős";

        break;

    case SARKVIDEKI:

        tipusString = "Sarkvidéki";

        break;

    case MADARKETREC:

        tipusString = "Madárketrec";

        break;

}

tipusNode.setTextContent(tipusString);

elohelyNode.appendChild(teruletNode);
elohelyNode.appendChild(tipusNode);
elohelyek.appendChild(elohelyNode);
}

```



```
}
```

Adatlekérdezés:

```
package XMLTaskB00P5Y.DOMParseB00P5Y.hu.domparsed.B00P5Y;
```

```
import java.io.BufferedReader;
```

```
import java.io.File;
```

```
import java.io.IOException;
```

```
import java.io.InputStreamReader;
```

```
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.Node;
```

```
import org.w3c.dom.NodeList;
```

```
import org.xml.sax.SAXException;
```

```
public class DOMQueryB00P5Y {
```

```
    public static void main(String arg[])
```

```
        throws SAXException, IOException, ParserConfigurationException, XPathExpressionException {
```

```
        File inputFile = new File("XMLTaskB00P5Y/XMLB00P5Y.xml");
```

```

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

Document doc = dBuilder.parse(inputFile);
doc.getDocumentElement().normalize();

// Build XPath tree
XPath xPath = XPathFactory.newInstance().newXPath();

int inputCode = 0;
System.out.println("Menu: ");
System.out.println("1) - A hímnemű állatok neveinek kiírása");
System.out.println("2) - A 2018-ban és attól később született állatok lekérdezése");
System.out.println("3) - December havi nyitvatartás lekérdezése");
System.out.println("4) - Az utolsó 3 takarítás lekérdezése");
System.out.println("Más billentyű) - Kilépés");
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
String input = br.readLine();

try {
    inputCode = Integer.parseInt(input);
} catch (Exception e) {
    inputCode = 0;
}

String expression = "";

switch (inputCode) {
    case 1:
        expression = "//allat/nev[../nem='Hím']";

```

```

        break;
    case 2:
        expression = "//allat[number(translate(/szuletes-ideje,'-',')) > 20180101]";
        break;
    case 3:
        expression = "//nyitvatartas[@honap=12]";
        break;
    case 4:
        expression = "//takaritas[position() > last() - 3]";
        break;
    default:
        System.out.println("Bye");
        return;
}

NodeList nodeList = (NodeList) xPath.compile(expression).evaluate(doc, XPathConstants.NODESET);
for (int i = 0; i < nodeList.getLength(); i++) {
    Node node = nodeList.item(i);
    DOMReadB00P5Y.printNodeRecurse(node);
}
}
}

```