

LINQ feladatok – 3

LINQ to XML

A feladatokhoz használjuk a *Szemelyek.xml* fájlt. A gyökérelemben *Szemely* elemek találhatók, mint pl. az alábbi:

```
<Szemely>
  <Azon>1</Azon>
  <Nev>János</Nev>
  <Nem>férfi</Nem>
  <Telefon Tipus="Otthoni">123-456</Telefon>
  <Telefon Tipus="Munkahelyi">1-111-222</Telefon>
  <Lakcim>
    <IrSzam>8000</IrSzam>
    <Helyseg>Székesfehérvár</Helyseg>
    <Utca>Jancsár</Utca>
    <Hazzsam>10</Hazzsam>
  </Lakcim>
</Szemely>
```

Ha .NET alkalmazásokban XML adatokkal szeretnénk dolgozni, akkor a *LINQ to XML* több osztályt is kínál erre a célra:

XDocument: egy teljes XML-dokumentumot képvisel, beleértve az XML-deklarációt és a gyökérelemet is. Tartalmazhat továbbá XML dokumentumtípust (DTD) és feldolgozási utasításokat is. Az *XDocument* osztály metódusokat biztosít az XML-dokumentumok betöltésére és mentésére, valamint a dokumentum tartalmának lekérdezésére és módosítására.

XElement: egy XML-elemet képvisel egy XML-fán belül. Egy *XElement* tartalmazhat más *XElement* objektumokat, valamint *XAttribute* objektumokat. Az *XElement* osztály metódusokat biztosít egy elem attribútumainak és gyermekelemeinek lekérdezésére és módosítására, valamint új elemek és attribútumok létrehozására.

XAttribute: egy XML-elemhez kapcsolódó attribútumot képvisel. Egy *XAttribute* csak egyetlen *XElement*hez tartozhat. Az *XAttribute* osztály metódusokat biztosít egy attribútum értékének lekérdezésére és módosítására, valamint új attribútumok létrehozására.

1. a) Töltsük be (*XElement.Load*) és listázzuk ki a képernyőre az összes személy adatait az *XElement* osztály segítségével! Az *XElement* objektumok felsorolható gyűjteménye kerüljön a *szemelyek* nevű változóba!
b) Töltsük be (*Xdocument.Load*) és listázzuk ki a képernyőre az összes személy adatait az *XDocument* osztály segítségével! A gyűjtemény neve legyen *szemelyek2*.
(Ha a gyökérelemre nincs szükség, akkor használhatjuk az *xdocument.Root.Elements()* kifejezést.)
2. Listázzuk ki csak a neveket a *szemelyek* gyűjteményből!
3. LINQ lekérdező kifejezés formátummal gyűjtjük ki a férfiak adatait és írassuk is ki!
4. LINQ lekérdező kifejezés formátummal gyűjtjük ki a nők adatait, majd a neveket és a munkahelyi telefonszámokat írassuk is ki!

5. *LINQ* lekérdező kifejezés formátummal gyűjtsük ki azon személyek adatait, akik Pécsen laknak, majd írassuk ki a nevüket!
6. Listázzuk ki az összes irányítószámot! (Ez beágyazott elem.)
7. *LINQ* lekérdező kifejezés formátummal gyűjtsük ki és listázzuk az irányítószámokat növekvő sorrendben! Használjunk tartományváltozót (*let* használata)!
8. Adjuk meg az első kettő/a 2. és a 3./az utolsó kettő személy nevét!
9. Adjuk meg egy adott pozíción, például a 3. helyen lévő személy adatait!
10. Állítsuk elő az alábbi tartalmú *ujszemelyek.xml* fájlt!

```
<?xml version="1.0" encoding="utf-16"?>
<Szemelyek>
  <Szemely>
    <!--Most csak 3 elemmel-->
    <Azon>1</Azon>
    <Nev>Béla</Nev>
    <Nem>férfi</Nem>
  </Szemely>
</Szemelyek>
```