

## IMPERA

---

# PÉLDATÁR KÉSZÍTÉSE

Az Impera a feladatlapokat példatárak alapján állítja elő. A példatár olyan XML-fájl, amely a feladatkészlet mellett tartalmazza a feladatlap előállításának szabályait, és a dolgozat lebojnyolításának néhány paraméterét. Minden beszámolóhoz külön XML-fájlt kell létrehozni, amely elkészíthető XML-szerkesztővel vagy az Impera bővítménnyel létrehozott Word-dokumentum exportálásával.

## XML ALAPISMERETEK

Az XML az eXtensible Markup Language (bővíthető jelölőnyelv) rövidítése, amelynek segítségével az adatokat strukturáltan tárolhatjuk szövegfájlokban, így azok az emberek és a programok számára egyaránt értelmezhetők.

Az XML-dokumentumok deklarációkat, elemeket, feldolgozási utasításokat és megjegyzéseket tartalmazhatnak. Ezeknek a komponenseknek egy része kötelező, és vannak, amelyek opcionálisak. A dokumentum szerkezetére tekintsük az alábbi egyszerű mintát:

```
<?xml version="1.0" encoding="UTF-8"?>
<könyv kiadás="magyar">
  <szerező>Ken Follett</szerező>
  <cím>A tűzoszlop</cím>
  <ISBN>978 963 406 506 7</ISBN>
</könyv>
```

Az első sor az ún. XML-deklaráció, amely a jelölőnyelv verzióját és a fájl karakterkódolását specifikálja. Ezt követi a dokumentumelem (más néven gyökérelem), amelyből az XML-dokumentumokban csak egy lehet. Az elemeket nyitó- és zárócímkékkkel adjuk meg, tartalmuk lehet szöveg, további elemek, illetve ezek keveréke. A mintában látható cím egy szöveges elem. Az elemek nyitórésze jellemzőket is tartalmazhat `név="érték"` szintaxissal.

**MEGJEGYZÉS** Az üres elemeket rövidített formában is megadhatjuk, például `<cím/>`.

Az XML-dokumentum elemei tetszőleges mélységig egymásba ágyazhatók, szerkezetük egy fastruktúrát alkot.

Az XML-fájlok előállításához, módosításához használhatunk egyszerűbb szövegszerkesztőket is, mint a Notepad++, de érdemes inkább olyan eszközt választani, amely a szintaktikai kiemelésen túl további szolgáltatásokat is nyújt, például kódkiegészítést, érvényesség-ellenőrzést, struktúraspecifikus keresést/cserét, kifejezések végrehajtását stb. A későbbi munkálatok elvégzéséhez a kereskedelmi szoftverek közül az egyik legjobb az Oxygen XML Editor

(30 napos licenc kérhető a kipróbáláshoz), de számos ingyenes fejlesztőeszközt is alkalmas a feladatra, pl. Visual Studio, Visual Studio Code, Eclipse.

## EGYEDEK (ENTITÁSOK)

Az XML-dokumentumokban néhány karakter speciális jelentéssel bír, például a < a címkék kezdetét jelöli, ezért a szövegben ezek helyett egyedhivatkozásokat kell használni. Az előre definiált egyedhivatkozások az alábbiak:

&lt;	<	kisebb mint
&gt;	>	nagyobb mint
&amp;	&	et jel
&apos;	'	apoztróf
&quot;	"	idézőjel

A karakteregyedek számkóddal is megadhatók, például az Ω hexadecimálisan &#x3A9;, decimálisan &#937;.

## JÓLFORMÁZOTTSÁG

A programok az XML-fájlokat csak akkor tudják feldolgozni, ha megfelelnek az alábbi jólformázottsági követelményeknek:

1. A prolognak, amely az XML-deklarációt és a dokumentumelem típuspecifikációját tartalmazhatja, a dokumentum elején kell szerepelnie. Ezek megadása opcionális.
2. A dokumentumnak pontosan egy dokumentumelemet kell tartalmaznia.
3. Minden elemnek rendelkeznie kell zárócímkével.
4. Az XML-címkék megkülönböztetik a kis- és nagybetűket. A nyitó- és zárócímkéknek betűhelyesen egyezniük kell.
5. Az elemek nyitó- és zárócímkéjének ugyanazon elembe kell elhelyezkedniük. Például hibás az <f><d>szöveg</f></d>, mert a d elem zárórésze nem az f elembe van.
6. A jellemzőket az elem nyitócímkéjében kell megadni. Egy jellemző névből és értékből áll, ahol a név megelőzi az értéket, és a kettőt egy egyenlőségjel választja el egymástól. Két jellemző neve nem lehet azonos, az értéket idézőjelekben adjuk meg.
7. Az elemek belsejében és a jellemzőértékekben a különleges jelentésű karakterek helyett – mint pl. a <, >, &, " és a ' – egyedhivatkozásokat kell használni.
8. A megjegyzések szintaxisa a következő: <!-- szöveg -->. A megjegyzésen belül két kötőjel megadása nem megengedett.

## NÉVTEREK

Az elemeket névterekbe csoportosítva különíthetjük el egymástól. A névterek minősített elemneveket használnak (pl. `xml:space`), azaz a név előtt előtagként a névtér neve is szerepel, ezáltal két azonos nevű, de különböző előtaggal rendelkező elem megkülönböztethetővé válik. A névtér deklarálásának szintaxisa `xmlns:névtér_név="URI"`, például

```
xmlns:xml="http://www.w3.org/XML/1998/namespace"
```

A névterek használata nemcsak a névkonfliktusok elkerülése végett hasznos, arra is lehetőséget nyújt, hogy az XML-feldolgozó az eltérő feldolgozási célokra szánt elemeket egymástól megkülönböztethesse.

## ÉRVÉNYESSÉG-ELLENŐRZÉS

Séma megadásával előírhatjuk az XML-dokumentumban használható jelölőelemeket, azok struktúráját, illetve az értékekre vonatkozó korlátozásokat. A professzionális XML-szerkesztők a séma alapján ellenőrizni tudják a dokumentum érvényességét, valamint a kódkiegészítő funkciójuk révén környezettől függően ajánlják fel az adott helyre beszúrható elemek és attribútumok nevét, illetve azok értékét.

## XPATH

Az XPath nyelv segítségével kiválaszthatjuk az XML-dokumentum egyes részeit, vagy egyszerűbb műveleteket végezhetünk azokon (például számlálás, összegzés). Az XPath a dokumentum fastruktúrájában hét ún. csomóponttípust különböztet meg:

1. gyökér (root)
2. elem (element)
3. jellemző (attribute)
4. szöveg (text)
5. megjegyzés (comment)
6. feldolgozási utasítás (processing instruction)
7. névtér (namespace)

Az XPath fának egyetlen gyökércsomópontja van, ez tartalmazza a fa összes többi csomópontját (például a dokumentumelemet). A szülőcsomópont magába foglalja a gyerekcsomópontokat. A jellemző- és a névtércsomópontok információt tartalmaznak a szülőcsomópontról.

Minden csomópont rendelkezik sztring reprezentációval (helyi név), illetve egyes csomópontoknak további jellemzője az ún. kiterjesztett név, amely a névtér nevéből és a helyi névből áll. (Ezek az egyes csomóponttípusok esetén eltérő módon képződnek.)

Azt, hogy miként juthatunk el az egyik csomópontból egy másikba az XPath fán, elérési út segítségével adhatjuk meg. Ez a kifejezés elérési lépésekből áll, melyeket / karakter választ el egymástól, kiértékelésük balról jobbra történik. Az elérési lépés három összetevőt tartalmazhat: irány, csomópontvizsgálat és predikátum.

Az alábbi példa két lépésből áll. Az első kiválasztja a `feladatlap` nevű gyerekelemeket, a második ezek utolsó `feladat` elemét. A lépésekben az irányt `::` választja el a csomóponttól, a predikátum `[]`-ben található.

```
child::feladatlap/child::feladat[last()]
```

Az *irány* azt jelöli ki, hogy a környezeti csomóponthoz képest melyik csomópontokat kell bevonni a keresésbe. Az irányokat az alábbi táblázat foglalja össze. Ezek egy része előre, míg másik részük hátra felé mutató keresést ír elő meghatározva a csomópontok sorrendjét.

<b>Irány</b>	<b>Leírás</b>
<code>self</code>	Maga a környezeti csomópont.
<code>parent</code>	A szülőcsomópont, ha létezik.
<code>child</code>	Gyerekcsomópontok, ha léteznek.
<code>ancestor</code>	Ősök, ha léteznek.
<code>ancestor-or-self</code>	A környezeti csomópont és ősei
<code>descendant</code>	Leszármazottak, ha léteznek.
<code>descendant-or-self</code>	A környezeti csomópont és leszármazottjai.
<code>following</code>	A környezeti csomópontot követő csomópontok a leszármazottakat nem beleértve.
<code>following-sibling</code>	A környezeti csomópontot követő testvércsomópontok.
<code>preceding</code>	A környezeti csomópontot megelőző csomópontok az őseket nem beleértve.
<code>preceding-sibling</code>	A környezeti csomópontot megelőző testvércsomópontok.
<code>attribute</code>	Jellemző csomópontok.
<code>namespace</code>	Névtér csomópontok.

Minden iránynak van elvi csomóponttípusa, amely megadja, hogy milyen csomóponttípusok legyenek kiválasztva. Jellemző, illetve névtér irány esetén ez jellemző, illetve névtér típus, minden más esetben elem.

A *csomópontvizsgálat* az irány által meghatározott csomóponthalmazt szűkíti.

Csomópontvizsgálat	Leírás
*	Az elvi típussal megegyező csomópontok.
node ()	Típustól függetlenül az összes csomópont.
text ()	A szöveg típusú csomópontok.
comment ()	A megjegyzés típusú csomópontok.
processing-instruction ()	A feldolgozási utasítás típusú csomópontok.
csomópontnév	Az adott nevű csomópontok.

Az elérési út néhány esetben egyszerűsített szintaxissal is megadható.

Normál alak	Rövid alak	Leírás
child::feladat	feladat	A környezeti csomópont feladat nevű gyerekelemei.
attribute::pont	@pont	A környezeti csomópont pont nevű jellemzője.
self::node ()	.	A környezeti csomópont.
parent::node ()	..	A szülőcsomópont.
/descendant-or-self::node () /	//	Az összes csomópont a dokumentum gyökerétől indulva.

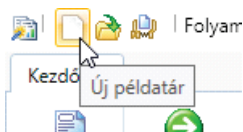
A *predikátum* tovább szűkítheti az eredményhalmazt, vagyis az irány és csomópontvizsgálat által meghatározott csomópontokból csak azok lesznek kiválasztva, amelyekre a feltételes kifejezés igaz értéket ad. A kifejezésekben literálok, csomópontok, változóreferenciák és függvényhívások szerepelhetnek operátorokkal összekapcsolva.

**MEGJEGYZÉS** A függvényekről a [https://www.w3schools.com/xml/xsl\\_functions.asp](https://www.w3schools.com/xml/xsl_functions.asp), az operátorokról a [https://www.w3schools.com/xml/xpath\\_operators.asp](https://www.w3schools.com/xml/xpath_operators.asp) oldalon talál részletes leírást.

## PÉLDATÁR KÉSZÍTÉSE XML-SZERKESZTŐVEL

### PÉLDATÁR LÉTREHOZÁSA

Új példatár a legegyszerűbben a Folyamat segítségével hozható létre. A gyorselérési eszköztárban kattintsunk az **Új példatár** parancsra, adjuk meg a fájl helyét és nevét, majd nyomjuk meg az **OK** gombot!



Az XML-fájl mellé a program elmenti a *peldatar.xsd* és *xml.xsd* fájlok legfrissebb változatait, a példatár érvényesség-ellenőrzése ezek alapján történik.

## A PÉLDATÁR FELÉPÍTÉSE

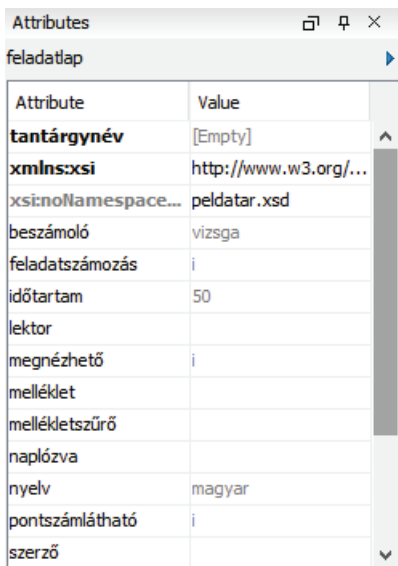
A példatárakat alapelemekből építjük fel, ezek háromféle csoportba sorolhatók. Az elvégzendő feladatokat *feladatleíró* elemekkel adhatjuk meg, mint például a bekezdés, az ábra, vagy a táblázat. A megoldásokat az *adatbekérő* elemek segítségével kérhetjük be, ide soroljuk a megjelölhető válaszokat, a különböző adatmezőket, lenyíló listát. A harmadik csoportba a *szerkezeti* elemek tartoznak, ezekkel a feladatlap-előállítás szabályait adhatjuk meg, illetve egységbe foglalhatjuk a feladatleíró és adatbekérő elemeket.

Az újonnan létrehozott példatár tartalma az alábbi:

```
<?xml version="1.0" encoding="utf-8"?>
<feladatlap tantárgynév=""
  xsi:noNamespaceSchemaLocation="peldatar.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

</feladatlap>
```

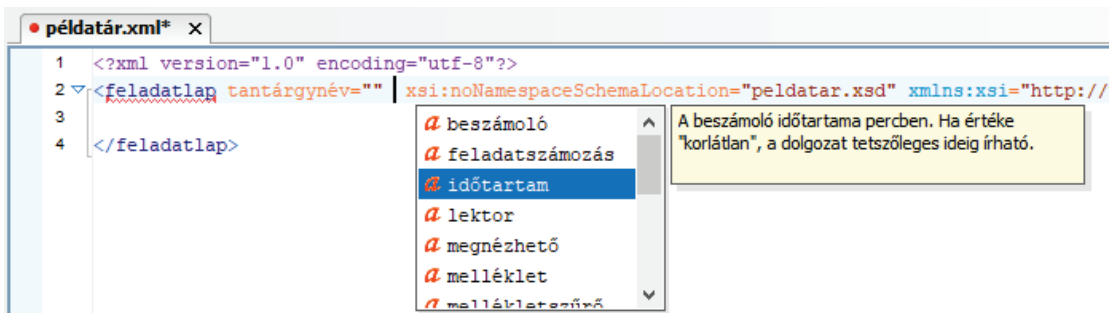
A dokumentumelem neve `feladatlap`, jellemzőinek értékét attribútumlistában állíthatjuk be, vagy az elem nyitócímkéjében begépeléssel adhatjuk meg.



Attribute	Value
<b>tantárgynév</b>	[Empty]
xmlns:xsi	http://www.w3.org/...
xsi:noNamespace...	peldatar.xsd
beszámoló	vizsga
feladatszámozás	i
időtartam	50
lektor	
megnézhető	i
melléklet	
mellékletszűrő	
naplózva	
nyelv	magyar
pontszámlátható	i
szerző	

A jellemzők egy része alapértelmezett értékkel rendelkezik, az ábrán ezeket szürke színnel látjuk. Az Oxygen XML Editor az explicit módon beállított jellemzőket a lista elejére teszi félkövér névvel, a többi alulra kerül, mindkét csoport név szerint rendezett. A címkében a jellemzőket tetszőleges sorrendben adhatjuk meg, akár több sorba is tördelhetjük őket.

Ha a program rendelkezik kódkiegészítéssel, gépelés közben az adott helyre beszúrható elemek vagy jellemzők nevei listából is kiválaszthatók. A sárga buborékban rövid leírást láthatunk az adott elemről.



A professzionális XML-szerkesztők folyamatosan végzik a dokumentum érvényesség-ellenőrzését, és hiba esetén piros aláhúzással jelzik a problémás helyeket. A gyökérelemet itt azért jelölte be a program, mert legalább egy elemet tartalmaznia kell.

## A FELADATLAP ELEM JELLEMZŐI

A feladatlapon elem jellemzőivel a tantárgy adatait, illetve a beszámoló megjelenítési és viselkedési beállításait adhatjuk meg. A tantárgyakat tantárgykóddal azonosítjuk. Egy tantárgyhoz több beszámoló tartozhat (vizsga, zárhelyik stb.). Amennyiben egy tantárgynak oktatónként eltérő a tananyaga, változat-azonosítószám megadásával tölthetünk fel különböző feladatösszeállításokat a beszámolókhöz.

Jellemző	Leírás
beszámoló	A beszámoló megnevezése.
feladatszámozás	Ha értéke i, a program sorszámozza a dokumentumelem feladat és feladatszámozás gyerekelemeit.
időtartam	A beszámoló időtartama percben, vagy korlátlan, ha a dolgozat tetszőleges ideig írható. Alapértelmezett értéke 50.
javítókulcs	Ha értéke i, gyakorlásnál megtekintéskor mutatja a javítókulcsot.
lektor	A feladatokat ellenőrző személy(ek) neve.
megnézhető	Ha értéke i, gyakorlásnál az értékelés után megtekinthető a javítás.
melléklet	A feltölthető melléklet maximális mérete MB-ban. A jellemző hiányában, nem adható be fájl a feladatlaphoz.
mellékletszűrő	A mellékletként feltölthető fájl típusa, ez alapján történik a fájl szűrése a <b>Megnyitás</b> ablakban. Például: Szövegfájlok .prn;*.txt;*.csv Szövegfájlok .prn;*.txt;*.csv Word-dokumentum .doc;*.docx A „Zip fájl *.zip” mindig hozzáadódik a megadott szűrőhöz.
naplózva	A naplózott folyamatok listája virgulával elválasztva.
nyelv	A feladatlapon nyelve.
pontszámlátható	Ha értéke i, a feladatok pontszáma megjelenik a feladatlapon.
tantárgykód	A tantárgy kódja.

Jellemző	Leírás
tantárgynév	A tantárgy neve.
szerző	A feladatokat készítő személy(ek) neve.
tiltva	A beszámoló alatt nem használható folyamatok listája.
verzió	A feladatösszeállítás verziója, amely lehet sorszám vagy dátum.
változat	A változat azonosítószáma.

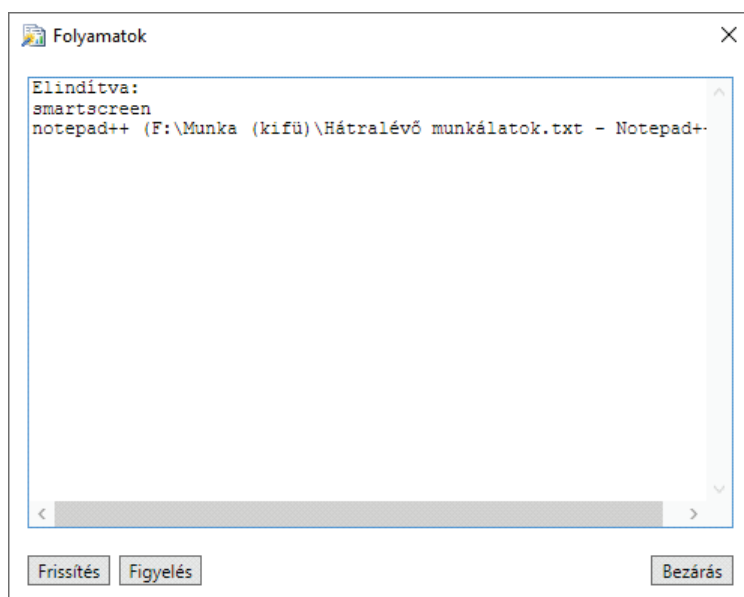
**MEGJEGYZÉS** A példatárakat a tantárgykód, a beszámoló és a változat jellemzők azonosítják. A fájlok szerverre való feltöltésekor az Intéző ezt a három adatot ellenőrzi, és ha ezek közül valamelyik nem egyezik meg, hibaüzenetet kapunk.

## ALKALMAZÁSOK TILTÁSA

A `tiltva` jellemzőben a nem engedélyezett folyamatokat sorolhatjuk fel virgulával elválasztva. Például a `chrome|firefox` érték esetén a Beszámoló a dolgozatírás közben leállítja a Chrome és Firefox böngészőket. A legtöbb esetben a feladatlap kitöltéséhez nincs szükség semmilyen alkalmazásra, ilyenkor a `minden` érték megadásával az összes folyamatot letiltathatjuk. Amennyiben néhány alkalmazás használható, tiltsunk le mindent, és `~` karakterrel kezdve adjuk meg az engedélyezett folyamatokat. Például a Total Commander futtatható a `minden|~totalcmd|~totalcmd64` értéknél. A fájlkezelő neve `explorer`, ne keverjük össze az Internet Explorerrel, ami az `iexplore` folyamat.

**MEGJEGYZÉS** A folyamattiltásnál a kis- és nagybetűk nincsenek megkülönböztetve.

A folyamatok nevét megtudhatjuk a Windows Feladatkezelőjéből vagy a Folyamat alkalmazás **Folyamatok listázása** parancsát használva. Az utóbbi módszernél a párbeszédablak-





ban a folyamat nevét a sorszám után olvashatjuk le. A név mögött zárójelben a folyamat ablakának felirata látható, ezek alapján könnyebb beazonosítani a keresett folyamatot. Ha a lista túl hosszú, nyomjuk meg a **Figyelés** gombot, indítsuk el (vagy állítsuk le) a programot, majd nyomjuk meg ismét a gombot! A figyelés bekapcsolása és kikapcsolása közötti változás alapján egyszerűen kideríthetjük a folyamat nevét.

**FONTOS** Vannak alkalmazások, amelyek több folyamatot is indítanak a használatuk során. Ilyen például a MatLab, amelynél a függvényszerkesztő egy külön folyamat. A Folyamatban a **Folyamatok tiltása** paranccsal leellenőrizhetjük, hogy a tiltólistánk megfelelő-e.

## NAPLÓZÁS

A naplózáshoz azoknak a folyamatoknak a nevét kell virgulával felsorolni, amelyekről jelentést szeretnénk kapni, ha mindegyikről kell, a minden értéket állítsuk be. A Beszámoló naplózni tudja az Excel makróit is, ehhez az `Excel-makró` kulcsszót is vegyük fel a listára.

A naplóbejegyzések száma alkalmazástiltással vagy a kivételek megadásával csökkenthető. Például a `minden|~explorer` értéknél a fájlkezelőkről nem készül jelentés. Kivételek megadásakor körültekintően járjunk el, mert például a szövegfájlok, PDF-ek tartalma fájlkezelővel is megjeleníthető.

**MEGJEGYZÉS** A naplózást akkor is érdemes bekapcsolni, ha minden alkalmazást letiltunk, ugyanis a Beszámoló nem állíthat le néhány folyamatot (pl. vírusirtók), amik akár böngészővel is rendelkezhetnek.

## SZERKEZETI ELEMEEK

Új példatár összeállításánál elsőként tervezzük meg a feladatlap felépítését, és ennek megfelelően hozzuk létre a főbb szerkezeti elemeket, illetve helyezzük el a szükséges tájékoztató szövegeket és címeket! Ha a csoportokat leírással látjuk el, később könnyebben tájékozódhatunk a dokumentumban.

A szerkezeti elemeket az alábbi táblázat ismerteti.

Elem	Leírás
<code>csoport</code>	A feladatlap előállításakor a gyerekelemek közül legfeljebb adott számú elem kerül kiválasztásra véletlenszerűen.
<code>blokk</code>	Rögzíti a gyerekelemek sorrendjét. A feladatlap előállításakor a blokk elemei együtt lesznek kiválasztva, egymáshoz viszonyított pozíciójuk nem változik.
<code>feladatblokk</code>	Egységbe zárja a feladatokat és a feladatleíró elemeket. Automatikus számozásnál a feladatblokk egy sorszámot kap, az abban lévő feladatok nem számozódnak.

Amikor nagy dokumentummal dolgozunk, az XML-szerkesztőkkel egyes elemek tartalmát összecukhatjuk, így csak azok a részek maradnak a fókuszban, amelyeket szerkeszteni kell. Az elemek kinyitott/összecukott állapotát egy kis háromszög (▼/▶) jelöli a bal oldali sávban. A váltáshoz egyszerűen kattintsunk az ikonra.

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  ▼ <feladatlap xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLoca
3  <bekezdés>A fájlok részére hozzon létre egy munkamappát saját nevével és neptunkódjával.
4  <!-- Témakörök: VBA (20 pont), Excel (x pont), Matlab (30-x pont) (összesen 50 pont) -->
5  <cim>Visual Basic for Applications</cim>
6  ▶ <csoport leírás="alapfogalmak egy adott forráskódrészletben (2 pont)"> [267 lines]
274 ▶ <feladat leírás="műveletek prioritása (2 pont)"> [23 lines]
298 <bekezdés>A programozási feladatok megoldásához szükséges segédsubrutinokat megtalálja a
299 ▼ <csoport leírás="első programozási feladat: ciklusszervezés, feltételírás (3 pont)">
300 ▼ <csoport leírás="feltételnek eleget tevő elemek száma">
301 ▶ <csoport leírás="adott intervallumbeli számok (százás, tízes helyi érték)"> [76 lines]
378 ▶ <csoport leírás="adott intervallumbeli számok (tízes, egyes helyi érték)"> [76 lines]
455 ▶ <csoport leírás="háromjegyű számok adott intervallumból (tízes, egyes helyi érték ha:
529 ▶ <csoport leírás="négyjegyű számok adott intervallumból (százás, tízes helyi érték ha:
603 ▶ <csoport leírás="adott intervallumbeli számok vizsgálata (oszthatóság, két feltétel)"
857 </csoport>
858 ▼ <csoport leírás="számelmélet">
859 ▶ <csoport leírás="Prim"> [159 lines]
1019 ▶ <csoport leírás="RelPrim"> [153 lines]
1173 </csoport>
1174 </csoport>
1175 ▶ <csoport leírás="második programozási feladat: tömbkezelés (4 pont)"> [421 lines]
1597 ▶ <csoport leírás="harmadik programozási feladat: Excel-adatok feldolgozása (4 pont)" kizár
3066 ▶ <csoport leírás="negyedik programozási feladat: összetett adatfeldolgozás (5 pont)"> [10
4110 <cim>Mérnöki számítások (Microsoft Office Excel 2010)</cim>
4111 <bekezdés típus="fontos">Az alábbi feladatok a Microsoft Excel 2010 magyar változatára va
4112 ▶ <csoport leírás="függvényábrázolás, célértékkeresés (6 pont)"> [201 lines]
4314 ▶ <csoport leírás="Solver (5 pont)"> [1487 lines]
5802 <cim>Mérnöki számítások (Matlab)</cim>
5803 ▶ <csoport leírás="egyenletrendszer + függvényábrázolás (10 pont)"> [2102 lines]
7906 ▶ <csoport leírás="ipari feladat (9 pont)"> [919 lines]
8826 </feladatlap>
    
```

Az ábrán látható összeállítás 1063 feladatot tartalmaz többszintű csoportosítással. A témaköröket a fő csoportok adják, az alcsoportokban az adott témakörbe tartozó, de eltérő jellegű feladatok vannak. A legbelső csoportok feladatai itt többnyire hasonlóak, sokszor csak a kiinduló adatok térnek el.

A csoportkialakítás többféle lehet. A legegyszerűbb eset az, amikor a feladatokat egybeömlesztjük, és abból választjuk ki a kívánt számút. A következő szintet az jelenti, amikor valamilyen szempont alapján több csoportot hozunk létre, például témakörök vagy leckék szerint. Ez biztosítja az egységesebb feladatlapok előállítását.

Ha ennél is bonyolultabb összeállítást szeretnénk, alkalmazhatunk kizárást a kiválasztás szabályozásához. Az `id` jellemzővel a csoportok azonosítóval láthatók el. Az adott azonosítójú csoport kiválasztását úgy akadályozhatjuk meg, hogy egy másik csoport kizárva jellemzőjében az azonosítót megadjuk.

```

<csoporth leírás="Első témakör">
  <csoporth kizárva="elmélet2 elmélet3"></csoporth>
  <csoporth kizárva="egyszerű2 egyszerű3"></csoporth>
  <csoporth kizárva="összetett2 összetett3"></csoporth>
</csoporth>
<csoporth leírás="Második témakör">
  <csoporth id="elmélet2" kizárva="elmélet3"></csoporth>
  <csoporth id="egyszerű2" kizárva="egyszerű3"></csoporth>
  <csoporth id="összetett2" kizárva="összetett3"></csoporth>
</csoporth>
<csoporth leírás="Harmadik témakör">
  <csoporth id="elmélet3"></csoporth>
  <csoporth id="egyszerű3"></csoporth>
  <csoporth id="összetett3"></csoporth>
</csoporth>

```

A példában 3 témakörhöz készítünk elméleti, egyszerű, illetve összetett számítási feladatokat, és a külső csoportok a feladatlap gyerekelemei. Az első témakörnél bármelyik feladat kiválasztható. A másodiknál olyan típusú már nem, amilyen az elsőnél ki lett választva, a harmadiknál pedig olyan nem, ami az első kettőnél.

**FONTOS** A kizárást befolyásolja az elemek feldolgozási sorrendje. Csoporton belül két testvércsoport kölcsönösen egymást is kizárhatja. Ilyenkor az előbb kiválasztott elem feldolgozásra kerül, a másik már biztosan nem. Ha ezt a másik elemet is fel kellett volna dolgozni, a szülőcsoport az előírtnál kevesebb elemet fog a feladatlagra helyezni.

A ciklus jellemző megadásával a kiválasztás elsősorban prioritási szintek alapján történik. Az első szinthez az alábbi gyerekelemek tartoznak:

- **nap:** a nap első órájában az első, a másodikban a második stb., és attól számítva minden 24. elem,
- **hét:** hétfőn az első, kedden a második stb., és attól számítva minden 7. elem,
- **hónap:** elsején az első, másodikán a második stb., és attól számítva minden  $n$ ., ahol  $n$  a hónap napjainak száma.

A második prioritási szintre a következő óra vagy nap elemei kerülnek, a harmadikba az azt követőé és így tovább. Ha a kiválasztandó elemek száma kevesebb, mint amennyi az első prioritási szinthez tartozik, akkor azok közül véletlenszerűen kerül kiválasztásra a szükséges számú, ha több, akkor az első szint összes eleme fel lesz dolgozva, a további elemek pedig a következő szintről kerülnek ki.

Az alábbi ábrán látható struktúrából 7.50-kor a Zene 1 csoportból kerül kiválasztásra a feladat, 8.10-kor a Mogyorógyűjtés 1 csoportból. Ha a db jellemző értékét 3-ra állítjuk, 13.25-kor az Adószedés 2, az Adószedés 3b és a Zene 1 csoportból sorsolódnak a feladatok.

```

<csoport ciklus="nap">
  <csoport leírás="Mogyorógyűjtés 1"> [21 lines]
  <csoport leírás="Mogyorógyűjtés 1b"> [21 lines]
  <csoport leírás="Mogyorógyűjtés 2"> [23 lines]
  <csoport leírás="Virágpor 1"> [21 lines]
  <csoport leírás="Adószedés 1"> [21 lines]
  <csoport leírás="Adószedés 2"> [21 lines]
  <csoport leírás="Adószedés 3b"> [59 lines]
  <csoport leírás="Zene 1"> [23 lines]
</csoport>

```

Célszerű a feladatokat kevesebb csoportba rendezni, mint a ciklus óráinak vagy napjainak száma, mert például az éjjel 1 órás feladatok sohasem lesznek felhasználva, ugyanis abban az időpontban általában nem zajlik beszámoló.


**FONTOS** A ciklus jellemzőnek nincs hatása, ha a db jellemző értéke mind.

## A FELADATOK KIALAKÍTÁSA


A feladatokat feladatléíró és adatbekérő elemekből építjük fel, és ún. konténerelemekbe helyezzük. Az elemstruktúra kialakítása függ a feladat összetettségétől, befolyásolja a sorsolást, illetve a sorszámcímkék (1. feladat, 2. feladat stb.) és pontszámok kijelzését. Az egységbezárást háromféle módon végezhetjük:

1. A feladat elembe helyezett gyerekelemek szegélyben jelennek meg. Az adatbekérők összecsatolhatók (részletesen lásd később), egy sorszámcímké jelenik meg, a szegély alatt pedig az elérhető maximális pontszám látható.


### 1. feladat

A forgalom.xlsx  munkafüzet alapján válaszoljon az alábbi kérdésekre!

---

 Melyik nap volt a legkisebb forgalom?

---

 Mennyi volt az összforgalom decemberben?

 Ft

2 pont

```


<feladat>
<bekezdés>A <letöltés forrás="forgaloml.xlsx" alias="forgalom.xlsx">forgalom.xlsx</letöltés> mu
<utasítás>Melyik nap volt a legkisebb forgalom?</utasítás>
<bekezdés><dátum>2020.12.07</dátum></bekezdés>
<utasítás>Mennyi volt az összforgalom decemberben?</utasítás>
<bekezdés><szám>172405</szám> Ft</bekezdés>
</feladat>

```


2. Ha a feladatleírást és a részfeladatokat `feladatblokk` elembe helyezzük, egy sorszám-címke jelenik meg, de a pontszámok külön-külön láthatók.

### 1. feladat

A `forgalom.xlsx` munkafüzet alapján válaszoljon az alábbi kérdésekre!

 Melyik nap volt a legkisebb forgalom?

1 pont

 Mennyi volt az összforgalom decemberben?

Ft

1 pont

```
<feladatblokk>
<bekezdés>A <letöltés forrás="forgaloml.xlsx" alias="forgalom.xlsx">forgalom.xlsx</letöltés> mu
<feladat>
  <utasítás>Melyik nap volt a legkisebb forgalom?</utasítás>
  <bekezdés><dátum>2020.12.07</dátum></bekezdés>
</feladat>
<feladat>
  <utasítás>Mennyi volt az összforgalom decemberben?</utasítás>
  <bekezdés><szám>172405</szám> Ft</bekezdés>
</feladat>
</feladatblokk>
```

Mivel a `feladatblokk` elembe `csoport` is elhelyezhető, a részfeladatok sorsolhatóvá válnak. Például az alábbi feladatnál az első részfeladathoz alternatíva lehet a legnagyobb forgalom, a másodikhoz más hónapok összforgalma.

```
<feladatblokk>
<bekezdés>A <letöltés forrás="forgalom.xlsx" alias="forgalom.xlsx">forgalom.xlsx</letöltés> mu
<feladat>
  <utasítás>Melyik nap volt a legkisebb forgalom?</utasítás>
  <bekezdés><dátum>2020.12.07</dátum></bekezdés>
</feladat>
<csoport>
  <feladat>
    <utasítás>Mennyi volt az összforgalom novemberben?</utasítás>
    <bekezdés><szám>207500</szám> Ft</bekezdés>
  </feladat>
  <feladat>
    <utasítás>Mennyi volt az összforgalom decemberben?</utasítás>
    <bekezdés><szám>172405</szám> Ft</bekezdés>
  </feladat>
</csoport>
</feladatblokk>
```

3. Külön számozhatjuk a részfeladatokat, ha azokat `blokk` elembe helyezzük. A sorsolás hasonlóan történhet, mint a feladatblokknál. Az összetett feladat kezdetét `cím` elemmel jelezhetjük a feladatlapon.

## Táblázatkezelés

A `forgalom.xlsx` munkafüzet alapján válaszoljon az alábbi kérdésekre!

### 1. feladat

 Melyik nap volt a legkisebb forgalom?

1 pont

### 2. feladat

 Mennyi volt az összforgalom decemberben?

Ft

1 pont

```
<cím>Táblázatkezelés</cím>
<csoport>
  <blokk>
    <bekezdés>A <letöltés forrás="forgalom1.xlsx" alias="forgalom.xlsx">forgalom.xlsx</letöltés> mu
    <feladat>
      <utasítás>Melyik nap volt a legkisebb forgalom?</utasítás>
      <bekezdés><dátum>2020.12.07</dátum></bekezdés>
    </feladat>
    <feladat>
      <utasítás>Mennyi volt az összforgalom decemberben?</utasítás>
      <bekezdés><szám>172405</szám> Ft</bekezdés>
    </feladat>
  </blokk>
  ...
</csoport>
```

## FELADATLEÍRÓK

**Cím** • Összetettebb, például több témakört érintő dolgozatoknál a címekkel több szakaszra oszthatjuk a feladatlapot. A `cím` elem csak a dokumentumelemben helyezhető el.

**Utasítás** • Ezzel az elemmel kiemelve, sötét háttérrel és narancs kérdőjellel jeleníthetjük meg a kérdéseket, illetve az utasításokat. Csak a `feladat` elembe helyezhető el.

**Bekezdés** • A feladatlírások folyószövegét bekezdésekkel adjuk meg. Ha valamire fel szeretnénk hívni a figyelmet, használjuk a `típus` jellemzőt! A bekezdésekben elhelyezhető gyerekelemeket a **Hiba! A hivatkozási forrás nem található.** rész ismerteti.

Jellemző	Leírás
igazítás	Balra, közepre vagy jobbra igazíthatjuk a bekezdés szövegét.
sorköz	Módosíthatjuk a bekezdés sorközét. Értéke csak szimpla lehet.
típus	Fontos, megjegyzés vagy segítség típusú bekezdést hozhatunk létre.

**Ábra** • Kép az ábra elemmel helyezhető a feladatlapra. A `fájl` jellemzőben szereplő ábra eredeti méretben vagy a megadott szélességgel és magassággal jelenik meg. Arányos átméretezéshez elegendő a méretjellemzők egyikének megadása. Célszerű a képfájlokat a megjelenítési méretre kicsinyíteni, ha azok fájlmérete ezáltal csökkenthető. Ha az ábra teljes feladatlapszélességben nem olvasható, adjuk meg a fájlspecifikációt a `nagyfájl` jellemzőben is. Ilyenkor a képre kattintva eredeti méretben nézhető meg az ábra.

**FONTOS** Az elérési utakat az XML-fájlhoz relatívan kell megadni. A mappa- és fájlnevek csak az angol ábécé kisbetűit, számjegykaraktereket és aláhúzáskarakteret tartalmazhatnak!

Az Intéző és Beszámoló a képfájlokat csak akkor tölti le, amikor azokat meg kell jeleníteni a képernyőn. A feladatsorok csak az ábrák fájlspecifikációját tartalmazzák. Ha a példatárban egy képfájlt lecserélünk, a korábbi feladatlapokon is az új kép fog megjelenni.

Jellemző	Leírás
cím	Az ábra címe. Sorközi ábráknál buborékban jelenik meg.
fájl	Az ábra fájlspecifikációja.
igazítás	Balra, közepre vagy jobbra igazíthatjuk az ábrát.
magasság	Az ábra magassága pixelben.
nagyfájl	Annak a képnek a fájlspecifikációja, amely az ábrára kattintva jelenik meg eredeti méretben.
szélesség	Az ábra szélessége pixelben.

**Képlet** • A képletek a `képlet` elemben MathML, LaTeX és AsciiMath formátumban ad-

hatók meg. Például az  $\frac{(a+1) \cdot b}{c}$  az alábbi kódokkal jeleníthető meg:

```
<képlet>
<math xmlns='http://www.w3.org/1998/Math/MathML'>
  <mrow>
    <mfrac>
      <mrow>
        <mrow><mo>(</mo>
          <mrow>
            <mi>a</mi><mo>+</mo><mn>1</mn></mrow>
            <mo></mo></mrow><mo>&#x22C5;</mo><mi>b</mi></mrow>
          <mi>c</mi>
        </mfrac>
      </mrow>
    </math>
  </képlet>
```

```
<képlet>\frac{\left( a+1 \right)\cdot b}{c}\$</képlet>
```

```
<képlet>`((a+1)*b)/c`</képlet>
```

A MathML nehezebben szerkeszthető, mint a másik két formátum, több helyet igényel a tárolása, de megjelenítése a leggyorsabb, és vannak olyan képletek, amelyek csak ilyen módon adhatók meg. A LaTeX a műszaki gyakorlatban gyakran használt formátum, a Word és a MathType a képletobjektumokat LaTeX-re tudja váltani. Az AsciiMath a legegyszerűbb formátum, a <http://asciimath.org/> oldalon található a leírása.

Jellemző	Leírás
cím	A képlet címe, amely buborékban jelenik meg.
igazítás	Balra, középre vagy jobbra igazíthatjuk a képletet.

**MEGJEGYZÉS** A LaTeX és AsciiMath képleteket az *impera.lib* fájl *mathml* mappájában található scriptek konvertálják MathML formátumúra, megjelenítésük ezért lassabb.

**Táblázat** • A táblázatok többnyire léniákkal határolt oszlopokba rendezett adatsorok, azonban vonal nélkül elrendezéshez is használhatók. A táblázat elem jellemzőivel megadhatjuk a cellák alapértelmezett igazítását és a táblázat címét, kikapcsolhatjuk a rácsvonalazást, illetve háttérszínnel emelhetjük ki a páros számú adatsorokat.

Jellemző	Leírás
cellaigazítás	A cellák alapértelmezett igazítása.
cím	A táblázat címe.
rácsozás	Ha értéke <i>i</i> , a táblázat rácsvonalai láthatóak.
sorkiemelés	Ha értéke <i>i</i> , a páros számú adatsorok háttérszínnel lesznek kiemelve.

A táblázat sorai a címsor jellemző értéke alapján lehetnek címsorok vagy adatsorok.

Jellemző	Leírás
címsor	Ha értéke <i>i</i> , a sor címsor, egyébként adatsor.

A sorokba *cella* elemek helyezhetők. Ezek jellemzőivel beállíthatjuk a vízszintes és függőleges igazítást, a cella kiterjedését oszlopokban és sorokban mérve, a háttérszínt és az oszlopszélességet pixelben vagy a táblázat szélességének százalékában. A cellák igazítása alapértelmezetten balra történik. Ha az adatok zömét középre kell igazítani, akkor állítsuk a táblázat elem *cellaigazítás* jellemzőjét középre értékűre, így csak a balra zárt celláknál kell megadnunk az igazítás jellemzőt. A háttérszín értéke lehet CSS-színnév (például *red*), vagy *#RRGGBB* formátumú hexadecimális színkód, ahol az *RR* (vörös), *GG* (zöld) és *BB* (kék) egy tizenhatos számrendszerű egész szám 00 és FF között. Az oszlopszélességeket a böngésző a cellák adattartalma alapján beállítja. Ha az automatikus méret nem felel meg, az oszlop legfelső cellájában a *szélesség* jellemzővel az módosítható. A tükörszélességnél nagyobb táblázatoknál a vászon vízszintesen görgethetővé válik.



Jellemző	Leírás
figazítás	Függőlegesen fel, középre vagy le igazíthatjuk a cella tartalmát.
háttér	A cella háttérszíne.
igazítás	Vízszintesen balra, középre vagy jobbra igazíthatjuk a cella tartalmát.
oszlopegybe	A cella kiterjedése oszlopokban mérve.
soregybe	A cella kiterjedése sorokban mérve.
szélesség	A cella szélessége pixelben.

A cellákban elhelyezhető gyerekelemeket a **Hiba! A hivatkozási forrás nem található.** rész ismerteti.

**MEGJEGYZÉS** A feladat elemében lévő táblázatok adatbekérőket is tartalmazhatnak.

**Forráskód** • A forráskód elemmel a programok forráskódját monofonttal és szintaktikai kiemeléssel jeleníthetjük meg. A programnyelv a `nyelv` jellemzővel adható meg.

Jellemző	Leírás
<code>nyelv</code>	A programnyelv megnevezése.

A forráskódot karakteradatként adjuk meg, hogy az XML-értelmező ne távolítsa el a szóközisméltódéseket és a soremeléseket! A feladatlapon a tabulátor karakter hossza két szóköz.

```

11 ▾ <forráskód nyelv="Javascript"><![CDATA[// szám átalakítása szöveggé és előlnullázása adott hosszra
12 String.előlnull = function(szám, hossz) {
13     var s = String(szám);
14     while (s.length < hossz) s = '0' + s;
15     return s;
16 }];]></forráskód>
    
```

A fenti kód az alábbi módon jelenik meg:

```

// szám átalakítása szöveggé és előlnullázása adott hosszra
String.előlnull = function(szám, hossz) {
    var s = String(szám);
    while (s.length < hossz) s = '0' + s;
    return s;
};
    
```

A támogatott programnyelveket a <https://prismjs.com> honlap *Supported languages* része ismerteti.

**Felsorolás** • A felsorolás elemmel a felsorolt pontokat számozhatjuk, beűzhetjük vagy díszponttal láthatjuk el. Hogy mivel legyenek jelölve a felsorolás pontjai, a `típus` jellemző határozza meg. Az első pont sorszámát vagy betűjelét egy pozitív egész szám megadásával módosíthatjuk a kezdőérték jellemző segítségével.

Jellemző	Leírás
<code>kezdőérték</code>	Az első pont sorszámát vagy betűjelét előíró pozitív egész szám.
<code>típus</code>	Meghatározza, hogy mivel legyenek jelölve a felsorolás pontjai.

## SORKÖZI ELEMEEK

**Formázott szöveg** • Szöveget a `folyam` elemmel formázhatunk. Szedhetjük monofonttal vagy áthúzott betűkkel (`alak`), alsó vagy felső indexbe helyezhetjük (`index`), előírhatjuk a betűstílust (`stílus`), illetve a betűszínt CSS-szín névvel vagy `#RRGGBB` formátumú hexadecimális színkóddal (`szín`).

Jellemző	Leírás
<code>alak</code>	Szöveg szedése monofonttal vagy áthúzott betűkkel.
<code>index</code>	Szöveg alsó vagy felső indexbe helyezése.
<code>stílus</code>	A szöveg betűstílusa.
<code>szín</code>	A szöveg betűszíne. Értéke lehet CSS-szín név vagy <code>#RRGGBB</code> formátumú hexadecimális színkód

A gyakrabban használt formázásokat `folyam` helyettesítő elemekkel is elvégezhetjük: monofont (`m`), alsó index (`ai`), felső index (`fi`), dőlt (`d`), félkövér (`f`), félkövér dőlt (`fd`).

**Ábra és képlet** • A sorközi ábra és képlet jellemzőinek szerepe a korábban ismertetett illusztrációs elemekével megegyező.

**Jegyzet** • A jegyzet elem narancs betűszínnel jelenik meg a feladatlapon, a fölé húzott eger mutató kézzé változik. Ha rákattintunk a jegyzetre, a szöveg jellemzőiben megadott kiegészítés megjelenik (vagy eltűnik).

**New York** az Amerikai Egyesült Államok legnépesebb, az amerikai kontinens második legnagyobb városa. Központja a New York-i **agglomerációnak**, amely világviszonylatban is a legnépesebbek közé tartozik. Az USA és a Föld első számú, üzleti kereskedelmi, egészségügyi, oktatási, kutatási és fejlesztési,

**New York** az Amerikai Egyesült Államok legnépesebb, az amerikai kontinens második legnagyobb városa. Központja a New York-i **agglomerációnak** (nagyváros körüli településrendszernek, amelynek lakosai főképp a nagyvárosban dolgoznak, vásárolnak), amely világviszonylatban is a legnépesebbek közé tartozik. Az

Jellemző	Leírás
<code>szöveg</code>	A megjeleníthető/elrejtendő jegyzetszöveg.

A kiegészítő szöveg alapesetben az elem szövege mögé kerül, de virgulával módosíthatjuk a megjelenés helyét, ahogy az a fenti mintában is látható.

```
<jegyzet szöveg="nagyváros körüli településrendszernek, amelynek lakosai főképp
a nagyvárosban dolgoznak, vásárolnak">agglomerációnak|,</jegyzet>
```

**Letöltés** • A feladatokhoz mellékelt fájlok a `letöltés` elemmel menthetők a felhasználó számítógépére. Ha több hasonló feladatunk van különböző forrásfájlokkal, akkor az `alias` jellemzővel írhatjuk elő, hogy azonos névvel legyenek mentve a fájlok.

```
<bekezdés>Olvassa be az <letöltés forrás="matlab/A_01.dat" alias="A.dat">A.dat</letöltés> és
a <letöltés forrás="matlab/b_01.dat" alias="b.dat">b.dat</letöltés> fájlt, amelyek egy lineáris
egyenletrendszer együtthatómátrixát és a hozzá tartozó (inhomogén) oszlopvektort tartalmazzák!</b
```

Jellemző	Leírás
alias	Ezzel az azonosítóval történik a fájl mentése.
forrás	A letölthető fájl specifikációja.

**FONTOS** Az elérési utat az XML-fájlhoz relatívan kell megadni. A mappa- és fájlnev csak az angol ábécé kisbetűit, számjegyekaraktereket és aláhúzáskaraktert tartalmazzon!

**Szószedet** • A szószedet elem egy narancs betűszínnel megjelenő szöveget eredményez, amelynél magyarázó szöveg jelenik meg buborékban, ha fölé húzzuk az egeret.

technológiai, média, művészeti és szórakoztatóipari központja. Itt található az ENSZ, és a világ legnagyobb bankjainak a székhelye is. A világ egyik leglátogatottabb városa és a legtöbb GÉP-t termelő város is egyben. Igen sokrétű és világhírű kulturális élete miatt a Föld kulturális fővárosaként is szokás emlegetni. Egyike a világ négy nagy divatfővárosának. A globális zeneipar és a nemzetköz  
bruttó hazai termék  
legfontosabb központja.

Jellemző	Leírás
leírás	A magyarázó szövege, amely buborékban jelenik meg.

**Sortörés** • Az új sor elemet akkor használjuk, ha új sorban akarjuk kezdeni a szöveget, de nem szeretnénk másik bekezdésbe helyezni.

## ADATBEKÉRŐK

A feladatok megoldásainak „begyűjtése” az adatbekérő elemekkel történik, ezek mindegyike a feladat elem leszármazottja az XML-dokumentumban. A bekérők alapesetben egymástól függetlenek, értékelésük külön-külön történik, de lehetőségünk van azok összeláncolására a feladat elemen belül. A láncolásnak két célja lehet:

1. Csak akkor adható pont a megoldásért, ha az összetartozó adatok mindegyike helyes. Például egy kitöltendő táblázatot soronként szeretnénk pontozni, vagy akkor fogadjuk el a feleletválasztás eredményét, ha azt szövegesen indokolják.
2. A helyes adatokért tört pontszámot szeretnénk adni. Például 8 számadatért 4 pont jár.

Az összeláncolt elemekre adható pontszámot az első elemnél kell feltüntetni. Értékeléskor az adatbekérők súlya azonos, ha a pontszámot szét kell közöttük osztani.

Mindegyik adatbekérő rendelkezik az alábbi jellemzőkkel:

Jellemző	Leírás
büntetés	A helytelen megoldásért levonható maximális pontszám.
csatolás	Az összeláncolt adatbekérők legelső eleménél akkor adjuk meg, ha a pontszámot szét akarjuk osztani közöttük. Értéke csak osztott lehet.
pont	A helyes megoldásért adható maximális pontszám. Ha értéke csatolt, az adatbekérőt a feladat elem megelőző bekérőjéhez láncoltuk.

**MEGJEGYZÉS** Ha az összeláncolt bekérők első eleménél a `pont` jellemző értéke `csatolt`, a láncolat 1 pontot ér.


Az adatbekérők egy része *önépítő*, ami azt jelenti, hogy `csoport` és `blokk` is lehet a gyerekelemük, így ezek végső összeállítása a feladatlap generálásakor alakul ki. Az önépítő adatbekérők többségénél a gyerekelemek feldolgozási módját a `sorrend` jellemző szabályozza. A megadható értékeket és azok hatását a következő táblázat foglalja össze.

### Jellemzőérték Leírás

állandó	A feldolgozás előfordulási sorrendben történik. Ha pl. minden gyerekelem <code>válasz</code> , azok mindig ebben a sorrendben kerülnek a feladatlapra. Ennek a beállításnak nincs látható eredménye, ha a gyerekelem egyetlen <code>csoport</code> elem.
újrakevert	A feldolgozás két menetben zajlik: 1. az elemeket előfordulásuk sorrendjében járjuk be, és az előfeldolgozás eredménye egy átmeneti tárolóba kerül, 2. a tároló elemei véletlenszerű sorrendben kerülnek a kimenetre. Ez a beállítás csak akkor szükséges, ha pl. a jelölendő és nem jelölendő válaszokat külön csoportba helyezünk, és a kisorsoltakat össze szeretnénk keverni.
változó	Az elemek feldolgozása véletlenszerű sorrendben történik.

## VÁLASZOK

Ezzel az adatbekérővel válaszlehetőségeket jeleníthetünk meg a képernyőn, amelyek közül a felhasználónak ki kell választania a megfelelő(ke)t. A válaszlehetőségek előtt alapesetben rádiógomb jelenik meg, ha közülük pontosan egyet szükséges megjelölni, egyébként jelölőnégyzet. Az alapértelmezett beállítás jellemzőkkel módosítható. Az alábbi ábrán látható feladatnál az „aktuális paraméter” a helyes megoldás, ennek ellenére jelölőnégyzetek vannak a sorok elején. Az utolsó válaszlehetőség mindig alulra kerül, sosem keveredik a válaszok közé. Azt akkor kell bejelölni, ha a felette található válaszlehetőségek egyike sem helyes.

 Az alábbiak közül melyik fogalom „található meg” a programkódban?

```

For i = 1 to 5
  MsgBox i & " db"
Next

```

- metódus
- típus
- aktuális paraméter
- explicit típusátalakítás
- A felsorolt lehetőségek egyike sem.

2 pont

```

<feladat>
  <utasítás>Az alábbiak közül melyik fogalom „található meg” a
  programkódban?</utasítás>
  <forráskód nyelv="visual-basic"><![CDATA[For i = 1 to 5
  MsgBox i & " db"
Next]]></forráskód>
  <válaszok pont="2" megjelenés="négyzet" egyiksem="i">
    <válasz>metódus</válasz>
    <válasz>típus</válasz>
    <válasz jelölt="i">aktuális paraméter</válasz>
    <válasz><szószedet leírás="kifejtett, nyelvi kifejezéssel
    jelölt">explicit</szószedet> típusátalakítás</válasz>
  </válaszok>
</feladat>

```

A válaszok elem jellemzői az alábbiak:

Jellemző	Leírás
egyiksem	Ha értéke <i>i</i> , utolsóként mindig megjelenik <i>A felsorolt lehetőségek egyike sem</i> . válaszlehetőség.
egyiksemválasz	<i>A felsorolt lehetőségek egyike sem</i> . válaszlehetőség szövege a megadott értékre módosítható.
megjelenés	Szabályozza, hogy a válaszok előtt rádiógombok vagy jelölőnégyzetek jelenjenek-e meg. Értéke automatikus vagy négyzet lehet.
részpont	Meghatározza a pontozási algoritmust. Értéke arányos, mérleg vagy nincs lehet.
sorrend	Meghatározza a gyerekelemek feldolgozásának módját a feladatlap előállításakor. Értéke állandó, újrakevert vagy változó lehet.

Azt, hogy egy választ jelölni kell-e, a válasz elem `jelölt` jellemzőjével adhatjuk meg.

Jellemző	Leírás
jelölt	Ha értéke <i>i</i> , a választ meg kell jelölni a helyes megoldáshoz.

A válaszok elem *önépítő*, vagyis a feladatlpra kerülő válaszlehetőségek nagyobb halmazból is sorsolhatók. A generáláskor előállított összeállításban a jelölendő elemek száma példányonként eltérhet, és emiatt a megjelenés is változhat. Amennyiben azonos vezérlőt szeretnénk minden esetben, állítsuk be az egyiksem és/vagy megjelenés jellemzők értékét *i*-re, illetve négyzet-re.

Jelölendő válaszok száma	megjelenés jellemző	Alkalmazott vezérlő
0 vagy 1	automatikus	rádiógomb
0 vagy 1	négyzet	jelölőnégyzet
> 1	automatikus	jelölőnégyzet
> 1	négyzet	jelölőnégyzet

Kitöltöttnek számít a bekérő, ha legalább egy válasz meg van jelölve. A kitöltetlen bekérő 0 pontot ér, egyébként az értékelés a részpont jellemző értéke alapján az alábbiak szerint történik:

**nincs:** Maximális pontot ér a megoldás, ha minden elem be van jelölve, amit kell, és egy sem, amit nem, ellenkező esetben a büntetésnek megfelelő pontlevonás jár érte.

**arányos:** A kapott pontszám a helyesen megjelölt elemek számával arányos, de helytelen jelölés esetén a megoldásért a büntetésnek megfelelő pontlevonás jár. Ha például 3 helyes válasz van, és a hibátlan megoldás 6 pontot ér, akkor egy helyes válasz bejelöléséért 2 pont jár, 2 helyes válaszáért 4.

**mérleg:** A pontszám a helyesen és helytelenül megjelölt válaszok különbségével arányos. Negatív eredménynél a pontlevonás nem lehet nagyobb a büntetés értékénél.


**FONTOS** A részpont kiszámítása a megjelölendő válaszok száma alapján történik.

**MEGJEGYZÉS** Amennyiben a részpont tört, az eredmény annak egész része lesz. Ha nincs megadva a büntetés jellemző, akkor értéke 0.

Azoknál a feladatoknál, ahol 4 válaszlehetőség közül 1 helyes, a várható érték 25%, vagyis a vakon kitöltött feladatok pontszámának negyedét megkapjuk. Ez az érték 0%-ra módosítható, ha például 3 pont jár a helyes megoldásért és -1 a hibásért.

## ÁLLÍTÁSOK

A bekérővel állításokat helyezhetünk a feladatlpra, amelyekről el kell dönteni, hogy azok igazak-e vagy sem. A válaszok lenyíló listákkal adhatók meg.

 Döntse el, hogy az alábbi állítások igazak vagy hamisak!

A számlát nem minden esetben lehet nyugtával helyettesíteni.

A teljesítés időpontját nem kell feltüntetni, csak a számlakibocsátását.

1 pont

```
<feladat>
  <utasítás>Döntse el, hogy az alábbi állítások igazak vagy
    hamisak!</utasítás>
  <állítások pont="1">
    <állítás érték="i">A számlát nem minden esetben lehet nyugtával
      helyettesíteni.</állítás>
    <állítás érték="h">A teljesítés időpontját nem kell feltüntetni,
      csak a számlakibocsátását.</állítás>
  </állítások>
</feladat>
```

Az állítások elem jellemzői az alábbiak:

Jellemző	Leírás
részpont	Meghatározza a pontozási algoritmust. Értéke arányos, mérleg vagy nincs lehet.
sorrend	Meghatározza a gyerekelemek feldolgozásának módját a feladatlap előállításakor. Értéke állandó, újrakevert vagy változó lehet.

Az állítások igazságértéke az érték jellemzővel adható meg.

Jellemző	Leírás
érték	Értéke $i$ , ha az állítás igaz, és $h$ , ha hamis.

Az állítások elem önépítő, vagyis az állítások sorsolhatók. A gyerekelemek feldolgozása ugyanúgy történik, mint a válaszok elemnél.

Kitöltöttnek számít a bekérő, ha legalább egy állítás igazságértéke meg van adva. A kitöltetlen bekérő 0 pontot ér, egyébként az értékelés a részpont jellemző értékétől függ:

nincs: Maximális pontot ér a megoldás, ha minden állításnál helyes igazságérték van megadva, ellenkező esetben a büntetésnek megfelelő pontlevonás jár érte.

arányos: A kapott pontszám a helyes igazságértékek számával arányos. Ha a büntetés jellemző értéke pozitív, helytelen igazságértéknél pontlevonás jár a teljes megoldásért.

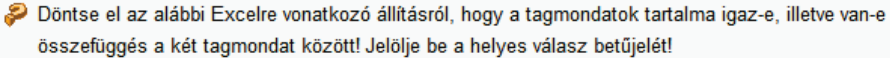
mérleg: A pontszám a helyes és helytelen igazságértékek különbségével arányos, negatív eredménynél a pontlevonás nem lehet nagyobb a büntetés értékénél.

**FONTOS** A részpont kiszámítása az állítások száma alapján történik.

Az 50%-os várható érték egy állításnál a pontszámmal azonos büntetés beállításával, több állítás esetén mérlegpontozással változtatható 0%-ra.

## RELÁCIÓK

A relációk elem összetett mondatainál azt kell eldönteni, hogy a tagmondatok igazak-e, illetve van-e ok-okozati összefüggés azok között.



**A:** Az első tagmondat igaz, a második is igaz, továbbá ok és okozati összefüggés van köztük.  
**B:** Az első tagmondat igaz, a második is igaz, de ok és okozati összefüggés nincs köztük.  
**C:** Az első tagmondat igaz, a második hamis.  
**D:** Az első tagmondat hamis, a második igaz.  
**E:** Mindkét tagmondat önmagában is hamis.

A kifejezések kiértékelésekor típusátalakításra is szükség lehet, mert a műveletek elvégzése csak az adott műveletre értelmezett típusokkal történhet.

**A B C D E**

1 pont

```

<feladat>
  <utasítás>Döntse el az alábbi Excelre vonatkozó állításról, hogy a
    tagmondatok tartalma igaz-e, illetve van-e összefüggés a két
    tagmondat között! Jelölje be a helyes válasz betűjelét!</utasítás>
  <relációk>
    <reláció érték="iiv">A kifejezések kiértékelésekor típusátalakításra is
      szükség lehet, mert a műveletek elvégzése csak az adott műveletre
      értelmezett típusokkal történhet.</reláció>
  </relációk>
</feladat>
    
```

A relációk elem jellemzői az alábbiak:

Jellemző	Leírás
részpont	Meghatározza a pontozási algoritmust. Értéke arányos vagy nincs lehet.
sorrend	Meghatározza a gyerekelemek feldolgozásának módját a feladatlap előállításakor. Értéke állandó, újrakevert vagy változó lehet.

A mondatok besorolása az érték jellemzővel adható meg.

Jellemző	Leírás
érték	A betűk megadják az első és a második tagmondat igazságértékét, illetve a közöttük lévő ok-okozati összefüggést. Például az <i>iin</i> azt jelenti, hogy mindkét tagmondat igaz, de nincs közöttük összefüggés.

Az relációk elem is önépítő, tehát a mondatok sorsolhatók. A gyerekelemek feldolgozása ugyanúgy történik, mint a válaszok elemnél.

Kitöltött a bekérő, ha legalább az egyik mondatnál van válasz. A kitöltetlen bekérő 0 pontot ér, egyébként az értékelés a részpont jellemző értékétől függ:

*nincs*: maximális pontot ér a megoldás, ha minden mondatnál helyes a válasz, ellenkező esetben a büntetésnek megfelelő pontlevonás jár érte.

*arányos*: A kapott pontszám a helyesen megadott válaszok számával arányos. Rossz válasz előfordulásakor a büntetésnek megfelelő pontlevonás jár a megoldásért, ha a büntetés jellemző értéke pozitív.

**FONTOS** A részpont kiszámítása a mondatok száma alapján történik.

## ADATMEZŐK


A feladatok megoldásait szövegben elhelyezett mezőkben is bekérhetjük. A különböző elemek közül az adat típusa, az ellenőrzés módja, illetve a mező viselkedése alapján választhatunk. Például a számadatok intervallum alapján is ellenőrizhetők a *szám* elemmel, a felsorolások, ragozott szavak pedig reguláris kifejezésekkel értékelhetők a legegyszerűbben.

A mezők hossza (a dátum kivételével) alapesetben a begépelt adattól függ, így azok méretéből nem lehet kikövetkeztetni a választ.



## Szöveg

Ezzel az elemmel szövegérték alapján történő összehasonlítással ellenőrizhető a megoldás. Az elsődlegesen elfogadható helyes választ az elem értékeként adjuk meg, a további választokat a szinonima jellemzőben sorolhatjuk fel virgulával (,) elválasztva.

 Egészítse ki az alábbi mondatot!

Származtatáskor a(z)  szerepe az, hogy definiálja az öt kibővítő osztályok számára az összes közös adatot és tagot.

1 pont

<feladat>

<utasítás>Egészítse ki az alábbi mondatot!</utasítás>

<bekezdés>Származtatáskor a(z)

<szöveg szinonima="szülőosztály|ősoosztály">alaposztály</szöveg> szerepe az, hogy definiálja az öt kibővítő osztályok számára az összes közös adatot és tagot.</bekezdés>

</feladat>

A szöveg elem jellemzői az alábbiak:

Jellemző	Leírás
betűérzékeny	Ha értéke i, értékeléskor megkülönböztetett a kis- és nagybetű.
elválasztó	A további válaszok elválasztó karaktere, ha a virgula nem megfelelő.
hossz	Előírja a mező hosszát karakterben.
javítás	Meghatározza a javítás módját, értéke felügyelt, gépi vagy kézi lehet.
szinonima	További elfogadható válaszok virgulával elválasztva.
szóköz	Meghatározza az üres karakterek kezelését a mező elhagyásakor.

A program a mező elhagyásakor, illetve a megoldás összegyűjtésekor az üres karaktereket a szóköz jellemző értéke alapján módosíthatja.

egybeír: minden üres karakter törlődik.

hámoz: a kezdő és záró üres karakterek törlődnek.

megtart: nem történik módosítás, minden üres karakter megmarad.

tisztáz: a kezdő és záró üres karakterek törlődnek, a szavak közé egy szóköz kerül.

Az értékelést a javítás jellemző értéke az alábbi módon befolyásolja:

felügyelt: ha a megoldás megegyezik az elsődleges válasszal vagy megtalálható a szinonimalistán, maximális pontot ér, különben manuálisan kell értékelni.

gépi: a pontozás a javítókulcs alapján történik. Ha a megoldás megtalálható az elfogadható válaszok között, maximális pontot ér, egyébként a büntetésnek megfelelő pontlevonás jár érte.

kézi: a választ manuálisan kell értékelni.

**MEGJEGYZÉS** A Beszámoló az elem értékét mutatja helyes válaszként a mező buborékjában. Ha ott a szinonimákat is meg kell jeleníteni, akkor azokat is az elem értékében soroljuk fel az elválasztó karakterrel!

### Szám

A megoldás ellenőrzése számérték alapján történik, vagyis az 1, a 01 és a 001 megegyezik.

Jellemző	Leírás
hossz	Előírja a mező hosszát karakterben.
előjel	Ha értéke i, a pozitív számok előjele is megjelenik a mezőben.
max	A legnagyobb elfogadható számérték.
min	A legkisebb elfogadható számérték.
tagolás	Ha értéke i, ezres tagolással jelenik meg a szám a mezőben.
tűrés	Az elfogadható legnagyobb eltérés abszolút értékben vagy százalékban negatív és pozitív irányban. Megadásakor a min és max jellemzők értékét a program figyelmen kívül hagyja.

A program a mező elhagyásakor ellenőrzi a megadott adatot. Ha az nem szám, pirosra állítja a betűszínt, ha igen, annyi tizedesjegyre kerekíti, amennyi a javítókulcsban szerepel.

**FONTOS** Ügyeljünk arra, hogy a javítókulcsként megadott szám a min és a max jellemzők értéke közé essen!

### Dátum

A dátum elemnél a megoldás ellenőrzése dátumérték alapján történik. A dátum évszámból, a hónap megnevezéséből vagy sorszámából, valamint a naptól áll. A dátum részeit az operációs rendszer területi beállításától függetlenül mindig ebben a sorrendben kell megadni.

A javítókulcs dátumában az évszám 2 vagy 4 számjegyű lehet, a hónap és a nap egy- vagy kétjegyű szám. Az évszám és hónap után kötelező kitenni a pontot, a nap után elhagyható. Például 99.01.12, 1985. 2. 13.

A mezőbe gépelt dátum formátumát formátumkóddal írhatjuk elő. Az évszám számjegyeit **é** jelöli, számuk kettő vagy négy lehet. A hónap megadására legalább egy, legfeljebb négy **h** betűt használunk (január esetén h: 1, hh: 01, hhh: jan., hhhh: január). A nap egy vagy két **n** betűvel jelölhető (hetedikénél n: 7, nn: 07).

Jellemző	Leírás
formátum	A dátum formátumkódja.

**MEGJEGYZÉS** A program a mező elhagyásakor vagy az adat kigyűjtésekor formázza meg a dátumot. Ha nincs megadva a formátum jellemző, a formázás a javítókulcs formátuma alapján történik. A hónap jelölésére használt római számot a program arabra cseréli.

## Általános mező

A mező elem ötvözi a szöveg és szám elemeket azáltal, hogy a megoldás ellenőrzése szöveg- és számérték alapján egyaránt elvégezhető. Ezt a bekérőt akkor alkalmazzuk, ha a feladatban különböző típusú adatokat kell megadni, és nem szeretnénk a számmezőkhöz szemantikai ellenőrzést. Az elem üresen hagyható, így olyankor is ezt a bekérőtípust használjuk, amikor a mezők egy részét nem kell kitölteni.

Jellemző	Leírás
betűérzékeny	Ha értéke <code>i</code> , szöveges adat értékelésekor megkülönböztetett a kis- és nagybetű.
előjel	Ha értéke <code>i</code> , a pozitív számok előjele is megjelenik a mezőben.
elválasztó	A további válaszok elválasztó karaktere, ha a virgula nem megfelelő.
hossz	Előírja a mező hosszát karakterben.
javítás	Meghatározza a szöveges adat javításának módját. Értéke <code>felügyelt</code> , gépi vagy kézi lehet.
max	A legnagyobb elfogadható számérték.
min	A legnagyobb elfogadható számérték.
szinonima	További elfogadható szöveges válaszok virgulával elválasztva.
szóköz	Meghatározza az üres karakterek kezelését a mező elhagyásakor.
tagolás	Ha értéke <code>i</code> , ezres tagolással jelenik meg a szám a mezőben.
típus	A megoldás értékelése a megadott típusnak megfelelően történik. Értéke <code>szám</code> vagy <code>szöveg</code> lehet.
tűrés	Az elfogadható legnagyobb eltérés abszolút értékben vagy százalékban negatív és pozitív irányban. Megadásakor a <code>min</code> és <code>max</code> jellemzők értékét a program figyelmen kívül hagyja.

## Ellenőrzés reguláris kifejezéssel

A mintaillesztés rugalmas és hatékony módszert kínál a megoldás értékelésére, ha az elfogadható válaszok száma túl sok vagy felsorolásuk nem lehetséges. A `regex` elemben egy, a `multiregex` elemben tetszőleges számú reguláris kifejezés adható meg az ellenőrzéshez.

Jellemző	Leírás
hossz	Előírja a mező hosszát karakterben.
javítás	Meghatározza a javítás módját. Értéke <code>felügyelt</code> vagy <code>gépi</code> lehet.
megoldás	A kézi javításkor a mező buborékjában megjelenítendő szöveg.
szóköz	Meghatározza az üres karakterek kezelését a mező elhagyásakor.

Az Impera .NET reguláris kifejezési mintákat használ, a szintaxisról a <http://aka.ms/regex> címen talál részletes leírást. A kifejezés teszteléséhez a <http://regexhero.net/> oldalról tölthet le programot.

Az alábbi adatbekérővel egy olyan mátrix méretét kérhetjük be, amely 4 soros, és nem lehet négyzetes. Az „egybeírás” miatt nem kell ügyelni a szóközökre (és azok számával), a számok között  $\times$ ,  $*$  és  $x$  is elfogadott.

```
<regexp szóköz="egybeír"  
  megoldás="pl. 4×8">^4[×*x] (?!4$) [1-9] \d*$</regexp>
```

Ennél az elemnél `regexp` gyerekelemekkel tetszőleges számú reguláris kifejezést adhatunk meg az ellenőrzéshez. A megoldás kizárólag csak akkor fogadható el, ha mindegyik kifejezés az `illeszkedés` jellemző értékének megfelelően illeszkedik vagy sem.

### Jellemző      Leírás

`illeszkedés`      Meghatározza, hogy a mintának illeszkednie kell-e a megoldásra.


A következő mintánál azt kell felsorolni, hogy funkciójuk alapján milyen csoportokba sorolhatók a Word mezői. A válaszban a szavak felcserélhetők, a felsorolásukhoz használható `sima` vagy pontos vessző, de azok el is hagyhatók. A megoldás akkor jó, ha az a megfelelő három szóból áll.

```
<multiregexp megoldás="eredményező, jelölőmező, akciómező" hossz="35">  
  <regexp>\b(?:eredmény(mező)?)\b</regexp>  
  <regexp>\b(?:jelölő(mező)?)\b</regexp>  
  <regexp>\b(?:akció(mező)?)\b</regexp>  
  <regexp>^\(\\b\\S+\\b[,;]?\\s*){3}$</regexp>  
</multiregexp>
```

**FONTOS** A mintaillesztésnél a kis- és nagybetű meg van különböztetve. Ha betűfüggetlen ellenőrzést szeretnénk, adjuk meg az `i` sorközi beállítást.

## LENYÍLÓ VAGY LÉPTETŐ LISTA

Lenyíló vagy léptető listát ad a szöveghez. A lenyíló lista rövid szöveges válaszoknál szövegmező helyett alkalmazva kiküszöböli az elgépelések, helyesírási hibák, illetve rossz ragozás okozta problémákat, a szövegkörnyezetből kiragadott mondatoknál támpontot ad a megoldáshoz. Hosszabb vagy formázott szövegek, képletek, ábrák mondatba történő behelyettesítéséhez használjunk léptető listát.


 Egészítse ki az alábbi mondatot!



Az XSLT  transzformációs nyelv.

1 pont

- univerzális
- objektum alapú
- objektumorientált
- procedurális
- deklaratív
- erősen típusos
- rekurziós
- gyengén típusos
- imperatív

```
<feladat>
  <utasítás>Egészítse ki az alábbi mondatot!</utasítás>
  <bekezdés>Az XSLT <lista>
    <elem helyes="i">deklaratív</elem>
    <elem>imperatív</elem>
    <elem>objektumorientált</elem>
    <elem>erősen típusos</elem>
    <elem>objektum alapú</elem>
    <elem>procedurális</elem>
    <elem>gyengén típusos</elem>
    <elem>rekurziós</elem>
    <elem>univerzális</elem>
  </lista> transzformációs nyelv.</bekezdés>
</feladat>
```

 **Egészítse ki az alábbi mondatot!**

Az elérési út egy olyan kifejezés, amely azt specifikálja,   milyen elérési lépéseket kell tenni a dokumentum aktuális csomópontjából a célcsomópont eléréséhez.

1 pont

```
<feladat>
  <utasítás>Egészítse ki az alábbi mondatot!</utasítás>
  <bekezdés>Az elérési út egy olyan kifejezés, amely azt specifikálja,
  <lista megjelenés="léptető">
    <elem>hogyan milyen irányba, milyen típusú csomópontokat kell kiválasztani
      a dokumentumfából.</elem>
    <elem>hogyan a környezeti csomópontokhoz képest mely csomópontokat kell
      bevonni a keresésbe.</elem>
    <elem helyes="i">hogyan történjen a navigáció az XPath fán az egyik
      csomóponttól egy másikig.</elem>
    <elem>hogyan érhető el a gyökércsomópont az aktuális
      kontextusból.</elem>
    <elem>milyen elérési lépéseket kell tenni a dokumentum aktuális
      csomópontjából a célcsomópont eléréséhez.</elem>
  </lista></bekezdés>
</feladat>
```

A lista elem jellemzői az alábbiak:

Jellemző	Leírás
megjelenés	A lista típusa. Értéke lenyíló vagy léptető lehet.
sorrend	Meghatározza a gyerekelemek feldolgozásának módját a feladatlap előállításakor. Értéke állandó vagy változó lehet.

A lista nem önépítő, gyerekelemei nem sorsolhatók. A listaelemnek egy jellemzője van:

Jellemző	Leírás
helyes	Megadja, hogy a listaelem elfogadható-e helyes megoldásként.


Ha egy feladatban több azonos tartalmú listára van szükség, az elemeket elhelyezhetjük egy elemlistában. Az alábbi kódrészlet egy ilyen `elemlista` elem látható, amely a feladat elem gyerekelemeként tetszőleges helyre tehető.

```
<elemlista id="lista">
  <elem>jövedelem</elem>
  <elem>bevétel</elem>
  <elem>költség</elem>
  <elem>-</elem>
</elemlista>
```

Az elemlista azonosítására az `id` jellemzőt használjuk, ezt kell megadni a lista `listaforrás` elemének `forrás` jellemzőjében. Az elemek a listában mindig az elemlistabeli sorrendben jelennek meg. A helyes megoldást a `helyes` jellemzőben az elem indexével adjuk meg, az első elemnél ez 1. Az alábbi kódrészletben a „bevétel” a helyes megoldás.

```
<lista><listaforrás forrás="lista" helyes="2"/></lista>
```

A képernyőn megjelenő feladat az alábbi ábrán látható.

 Jelölje az alábbi táblázatban, hogy az egyes mondatok mely fogalmak meghatározásai! Ha talál olyan meghatározást is, amelyikre egyik fogalom sem illik, jelölje „-” jellel!

A társas vállalkozásban személyesen közreműködő magánszemély tag által személyes közreműködése ellenében kapott juttatás (amit a társas vállalkozás költségei között elszámol)	bevétel
A bevételszerző tevékenységgel közvetlenül összefüggő, kizárólag a bevétel megszerzése, a tevékenység folytatása érdekében a naptári évben ténylegesen kifizetett, szabályszerűen igazolt fizetett összeg	jövedelem
Az a kapott vagyoni érték, amelyet az adóalany köteles visszaszolgáltatni	bevétel
Munkabér	költség
A munkáltató által elrendelt, a munkakör betöltéséhez szükséges ismeret megszerzését szolgáló képzés költsége	-

5 pont

**FONTOS** A `helyes` jellemző értéke a séma alapján nem ellenőrizhető, megadható az elemszámnál nagyobb index is. Ilyenkor a program minden megoldásra 0 pontot ad.

## RENDEZÉS

A rendezés adatbekérőnél A listaelemeket értékük alapján nemcsökkenő sorrendbe kell rakni Az elemek pozíciója billentyűzettel (SHIFT+ALT+FEL, SHIFT+ALT+LE), a nyomógombokkal, vagy egérrel történő húzással változtatható. A kitöltetlen feladatok sorszama szürke, színük az első elemmozdításakor változik narancssárgára.


A rendezés elem csak az általános adatbekérő jellemzőkkel rendelkezik. A rendezendő elemek értékét az `index` jellemző adja, ha azt az `érték` jellemzővel nem állítjuk be explicit módon.

Jellemző	Leírás
érték	Pozitív egész szám, amely meghatározza az elem helyét a listán. Ha nem adjuk meg, értéke megegyezik az <code>index</code> jellemzőjével.
index	Azonosítja az elemet. Értéke pozitív egész lehet.



A rendezés önépítő, vagyis a rendezendő elemek sorsolhatók. Az alábbi feladatnál a VBA műveletei közül 5 darab kerül a feladatlapra. A relációs operátoroknál látható `érték` beállítások jelzik az azonos prioritási szintet. (Ezek megadása a csoport jelenlegi beállításainál elhagyható, de amint két elemet sorsolunk, már szükséges.)

```
<feladat>
<utasítás>Rakja a műveleteket a végrehajtásuk szerinti sorrendbe!
Kezdje a legmagasabb prioritásával!</utasítás>
<rendezés pont="2">
  <csoporthatványozás (^)</elem>
  <elem index="2">negáció (-)</elem>
  <elem index="3">szorzás és osztás (*, /)</elem>
  <csoporthatványozás (^)</elem>
  <elem index="4">egész osztás hányadosa (\)</elem>
  <elem index="5">egész osztás maradéka (Mod)</elem>
</csoporthatványozás (^)</elem>
  <elem index="6">összeadás és kivonás (+, -)</elem>
  <elem index="7">karakterlánc-összefűzés (&amp;)</elem>
  <csoporthatványozás (^)</elem>
  <elem index="8">egyenlő (=)</elem>
  <elem index="9" érték="8">nem egyenlő (&lt;&gt;)</elem>
  <elem index="10" érték="8">kisebb (&lt;)</elem>
  <elem index="11" érték="8">nagyobb (&gt;)</elem>
  <elem index="12" érték="8">kisebb vagy egyenlő (&lt;=)</elem>
  <elem index="13" érték="8">nagyobb vagy egyenlő (&gt;=)</elem>
  <elem index="14" érték="8">Like</elem>
</csoporthatványozás (^)</elem>
  <elem index="15">logikai nem (Not)</elem>
  <elem index="16">logikai és (And)</elem>
  <elem index="17">logikai vagy (Or)</elem>
</csoporthatványozás (^)</elem>
</rendezés>
</feladat>
```

A lenti ábrán a sorsolás egyik végeredménye látható. Figyelje meg a szürke számokat, amely azt jelzi, hogy a feladat még érintetlen! Ha az elemek között vannak eltérő értékűek, akkor a kezdősorrendben biztosan változtatni kell a helyes megoldáshoz.

 Rakja a műveleteket a végrehajtásuk szerinti sorrendbe! Kezdje a legmagasabb prioritásával!

1. logikai nem (Not)
2. szorzás és osztás (\*, /)
3. negáció (-)
4. kisebb vagy egyenlő (<=)
5. összeadás és kivonás (+, -)

2 pont

## FOGD ÉS VIDD

A fogdésvidd adatbekérő fogalmak párosításához (a forráselemek és mezők száma megegyezik, és minden mezőhöz egy elem tartozik), csoportosításhoz (az elemek egy vagy több helyre tehetők, a mezőkbe több elem is kerülhet), illetve hozzárendeléshez (a forráselemek másolhatók, de minden mezőbe csak egy helyezhető el) használható. A forrásként megadott elemeket egér segítségével helyezhetjük a mezőkbe (nyelők).

A program automatikusan engedélyezi a forráselemek másolását, ha azok valamelyikét több helyre is el kell helyezni, illetve a mezőkbe több elemet tehetünk, ha valamelyik nyelvő gyerekelemeinek száma egynél nagyobb.

Jellemző	Leírás
halmaz	A mezőkbe több elem helyezhető, ha értéke <i>i</i> .
másol	A forráselemek több helyre tehetők, ha értéke <i>i</i> .
réspont	Meghatározza a pontozási algoritmust. Értéke arányos, mérleg vagy nincs lehet.

A fogdésvidd elem első gyerekeleme a forrás, ebben adhatjuk meg a forráselemeket.

Jellemző	Leírás
sorrend	Meghatározza a gyerekelemek feldolgozásának módját a feladatlap előállításakor. Értéke állandó, újrakevert vagy változó lehet.

A forráselemeket az *id* jellemzővel azonosítjuk.

Jellemző	Leírás
<i>id</i>	Azonosítja a forráselemet. Értéke pozitív egész.

A fogdésvidd elem második (és további) gyerekeleme bekezdés, táblázat vagy bekezdéseket tartalmazó csoport lehet. A nyelvők a bekezdésekben és a táblázatokban helyezhetőek el, ezekben *eref* elemekkel adjuk meg az oda tartozó forráselemeket.

Jellemző	Leírás
<i>id</i>	A hivatkozott elem azonosítója. Értéke egy létező forráselem-azonosító.

A fogdésvidd elem *önépítő*, a feladatlapra kerülő elemek, valamint a bekezdések és táblázatsorok sorsolhatók. A csoportkialakítástól függően előfordulhat, hogy a generálás után a nyelvők elemszáma példányonként változik, illetve véletlenszerűen kerülhet a feladatlapra olyan forráselem is, amely több nyelvőbe tartozik, ilyenkor állítsuk be a *halmaz* és/vagy *másol* jellemzők értékét *i*-re, hogy a feladat viselkedése minden esetben azonos legyen.

A nyelvők sorsolásakor azok a forráselemek törlődnek, amelyek kizárólag a kieső nyelvőkhöz tartoznak (vagyis a megmaradókhoz nem). Ha azt szeretnénk, hogy egy forráselem minden esetben megmaradjon, helyezzük mindig elembe. Az ún. kakukktojás-elemek, amelyeket

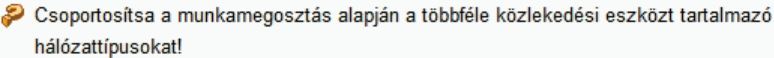



semmihez sem kell hozzárendelni, nem törlődnek. Az alábbi kódrészlet egy olyan feladathoz tartozik, ahol 10 ország fővárosát és államformáját kell megadni. Ha az országokból kisorsolunk négyet, a fővárosokból csak a hozzájuk tartozó 4 marad a forrásban, az államformákból viszont mindegyik, még ha némelyiket nem is kell felhasználni.





```
<forrás sorrend="állandó">
  <!-- főváros -->
  <elem id="101">Doha</elem>
  <elem id="102">Ammán</elem>
  <elem id="103">Damaszkusz</elem>
  <elem id="104">Taskent</elem>
  <elem id="105">Mogadishu</elem>
  <elem id="106">Rabat</elem>
  <elem id="107">Tripoli</elem>
  <elem id="108">Paramaribo</elem>
  <elem id="109">Vilnius</elem>
  <elem id="110">Minszk</elem>
  <!-- államforma -->
  <mindig>
    <elem id="201">monarchia</elem>
    <elem id="202">alkotmányos monarchia</elem>
    <elem id="203">köztársaság</elem>
    <elem id="204">szövetségi köztársaság</elem>
  </mindig>
</forrás>
```

Virgulával az elhelyezett elemek lerövidíthetők. Ha a szöveg egy virgulát tartalmaz, az előtte lévő szövegrész kerül a mezőbe, ha kettőt, a közöttük lévő.

```
<forrás sorrend="állandó">
  <elem id="1">horizontális| munkamegosztás</elem>
  <elem id="2">vertikális| munkamegosztás</elem>
</forrás>
```





Jelleg	Jellemzők
horizontális	a párhuzamos vonalvezetés érvényesül
vertikális	egy útvonalon csak egyfajta közlekedési eszköz jár
vertikális	például azonos (vagy nagyrészt azonos) útvonalon jár az autóbusz és a villamos
	alkalmazása akkor indokolt, ha a jelentkező utazási igényekhez egyedül egyik közlekedési eszköz sem elegendő
	minden vonalon az ottani forgalomnak leginkább megfelelő (kapacitású) eszközt alkalmazzuk
	több közlekedési eszköz együttes kapacitására van szükség
	a különféle közlekedési eszközök vonalai a végpontokon csatlakoznak egymáshoz

Kitöltöttnek számít a bekérő, ha legalább egy forráselemet mezőbe helyeztünk. A kitöltetlen bekérő 0 pontot ér, egyébként az értékelés a részpont jellemző értéke alapján az alábbiak szerint történik:

**nincs:** maximális pontot ér a megoldás, ha minden mezőben a megfelelő elemek vannak, ellenkező esetben a büntetésnek megfelelő pontlevonás jár érte.

**arányos:** a kapott pontszám a jó elemhelyezések számával arányos, de halmozáskor egyetlen rossz helyezésnél a megoldásért a büntetésnek megfelelő pontlevonás jár.

**mérleg:** a pontszám a jó és rossz elemhelyezések különbségével arányos. Negatív eredménynél a pontlevonás nem lehet nagyobb a büntetés értékénél.

**FONTOS** A nyelőkben lévő elemek sorrendje az értékelésnél nem számít. A részpont kiszámítása a mezőkbe helyezendő elemek száma alapján történik. Ha egy elemet két helyre kell tenni, akkor az kettőnek számít.

**MEGJEGYZÉS** Amennyiben a részpont tört, az eredmény annak egész része lesz. Ha nincs megadva a büntetés jellemző, akkor értéke 0.

## JELÖLŐ


Ezzel az elemmel jelölőnégyzetet helyezhetünk el a feladatlapon.

### Jellemző

### Leírás

jelölt

Ha értéke 1, a jelölőnégyzetet meg kell jelölni a helyes megoldáshoz.

 Jelölje a gazdasági eseménynél, hogy melyik vagyónrész, eredménytényező, a vagyon, illetve az eredmény hogyan változott!

Gazdasági esemény	Eszköz		Forrás		Költség		Ráfordítás		Bevétel		Eredmény	
	+	-	+	-	+	-	+	-	+	-	+	-
Áruértékesítés bevétele	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2 pont

## ESSZÉ

Az esszé adatbekérő legfeljebb 2000 karakteres szöveg vagy programkód bekérésére használható. A megoldások értékelése többnyire manuálisan történik, de reguláris kifejezések alapján is pontozhatjuk vagy szűrhetjük a megoldásokat (lásd `multiregexp` elem).

### Jellemző

### Leírás

mintaellenőrzés

Meghatározza a javítás módját reguláris kifejezések használatakor. Értéke elégséges vagy szükséges lehet.

típus

Az adat típusa. Értéke programkód vagy szöveg lehet. A programkód monofonttal jelenik meg, a tabulátor hossza 2 szóköz.

A kitöltetlen bekérő minden esetben 0 pontot ér. Ha nincs megadva reguláris kifejezés, az oktatónak kell értékelnie a megoldást, egyébként a mintaellenőrzés jellemző értékétől függően az alábbi történik:

elégleges: maximális pontot ér a válasz, ha megfelel mindegyik kifejezésnek, ellenkező esetben a büntetésnek megfelelő pontlevonás jár érte.

szükséges: a mintaillesztéssel kiszűrt hibás megoldásért a büntetésnek megfelelő pontlevonás jár, a javítókulcsnak megfelelő választ pedig manuálisan kell értékelni.

Az alábbi programozási feladatnál a helyes eredményt csak akkor fogadjuk el, ha mellékelik a forráskódot is. A forráskódra bonyolultabb megkötések is adhatók, itt mindössze csak azt írtuk elő, hogy legalább 50 karakteresnek kell lennie.

```
<feladat>
<utasítás>Írjon szubrutint az alábbi probléma megoldására!</utasítás>
<bekezdés>Hány olyan hárommal maradék nélkül osztható négyjegyű pozitív
egész szám van, amelyre a következő két állítás közül pontosan az egyik
igaz?</bekezdés>
<bekezdés><d>a)</d> A százas helyi értéken a 8 áll.<újsor/><d>b)</d>
Az egyes helyi értéken a 7 áll.</bekezdés>
<bekezdés>Eredmény: <szám pont="2">540</szám></bekezdés>
<bekezdés>Az Ön által írt szubrutin forráskódja:</bekezdés>
<esszé pont="csatolt" mintaellenőrzés="elégleges" típus="programkód">
<regex>.{50,}</regex>
</esszé>
</feladat>
```

Írjon szubrutint az alábbi probléma megoldására!

Hány olyan hárommal maradék nélkül osztható négyjegyű pozitív egész szám van, amelyre a következő két állítás közül pontosan az egyik igaz?

a) A százas helyi értéken a 8 áll.  
b) Az egyes helyi értéken a 7 áll.

Eredmény:

Az Ön által írt szubrutin forráskódja:

2 pont

## PROGRAMKÓD

A programkód adatbekérővel egy legfeljebb 2000 karakteres, tetszőleges .NET-nyelven írt forráskódot ellenőrizhetünk.

Jellemző	Leírás
nyelv	A forráskód programnyelve.

Az értékeléshez egy publikus, paraméter nélküli logikai függvényt kell készíteni `Teszt` névvel (az adott nyelven), amelynek értéke akkor igaz, ha a beadott megoldás az általunk előírt teszteseteknek megfelel. Például:

```
<programkód nyelv="CSharp"><![CDATA[public bool Teszt()
{
    return Darabszám(142, 200) == 7 && Darabszám(52, 103) == 5;
}]]></programkód>
```

Hiányzó vagy nem publikus `Teszt` függvény esetén manuálisan kell értékelni a megoldást. Szintaktikai hibánál nem dönthető el egyértelműen, hogy mi okozta a hibát, ezért – feltételezve, hogy a `Teszt` függvény helyes – a forrásprogramot rossznak minősíti a program.

**FONTOS** Szintaktikai hiba keletkezhet például akkor, ha a feladatul kitűzött szubrutin hibásan lett elnevezve vagy rossz a paraméterlistája. Ekkor a `Teszt` függvényt a fordítóprogram annak ellenére tartja hibásnak, hogy az valójában helyes.

Szerelvényreferenciák:

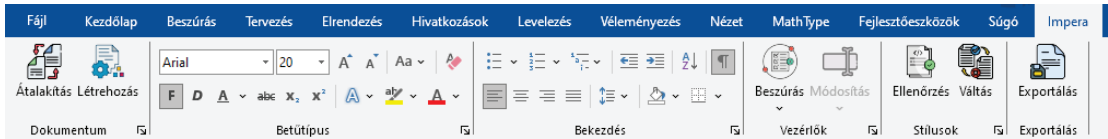
System.dll  
 System.Xml.dll  
 Microsoft.VisualBasic.dll (ha a nyelv Visual Basic)

Importált névterek:

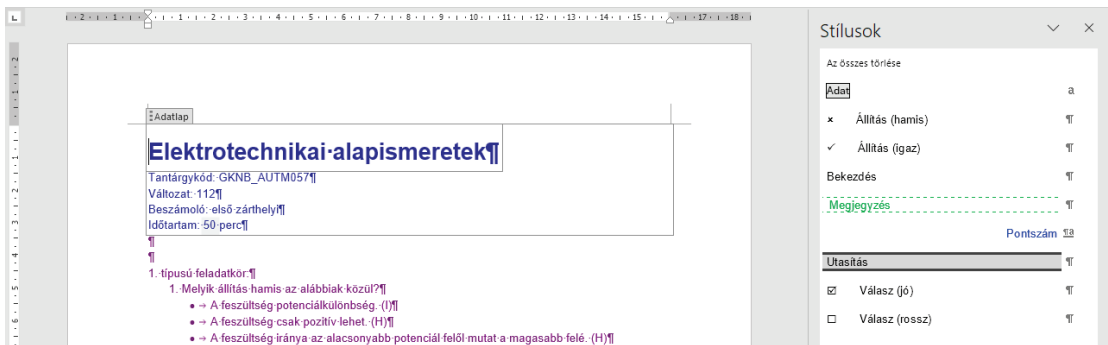
System  
 System.Collection  
 System.Collections.ObjectModel  
 System.Text  
 System.Text.RegularExpressions  
 System.Xml  
 System.Xml.Xsl  
 System.Xml.XPath  
 Microsoft.VisualBasic (ha a nyelv Visual Basic)

## PÉLDATÁR KÉSZÍTÉSE IMPERA BŐVÍTMÉNNYEL

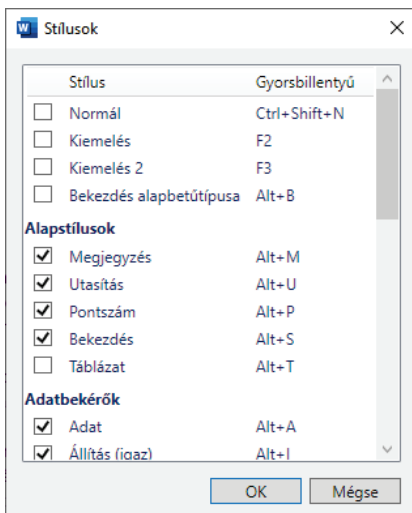
Az Impera bővítmény használatával feladatainkat Word-dokumentumban készíthetjük el. A szalag szinte mindent tartalmaz, ami az előállításához kell, ezért a legtöbb esetben nem is kell másokra átváltani.



A dokumentum szövegrészeinek szerepét stílusokkal és tartalomvezerlőkkel adjuk meg, így a munkálatok első fázisa a megfelelő stílusok elkészítése. Ehhez hozunk létre egy új üres dokumentumot vagy nyissunk meg egy meglévőt, és kattintsunk a **Létrehozás** parancsra!



A parancs elrejti azokat a stílusokat, amelyekre nincs szükség, létrehozza azokat, amelyeket a feladatok készítéséhez használhatunk, majd ezek közül a leggyakrabban használtakat megjeleníti a **Stílusok** panelen. A stílusok megjeleníthetők/elrejthetők a **Stílusok kezelése** ablakkal, de erre a célra jobb megoldást kínál a **Dokumentum** csoport párbeszédpanel-megjelenítő ikonjával előhozható **Stílusok** ablak.



A **Stílusok** panelen azok a stílusok lesznek elérhetőek, amelyek előtt pipa van a jelölőnégyzetben. A stílusokhoz a harmadik oszlopban látható billentyűkombinációk vannak hozzárendelve, ezek használatával gyorsan megformázhatjuk a szöveget.

A stílusok beállításai módosíthatók, ha azok nem felelnek meg számunkra. A **Létrehozás** parancs alkalmazásával az eredeti beállítások bármikor visszaállíthatók.

**MEGJEGYZÉS** Az **Átalakítás** parancs *Coedu* vagy *Impera* sablonnal elkészített dokumentumokat alakít át. Ha nem ilyen dokumentumon alkalmazzuk, a végeredmény megegyezik a **Létrehozás** parancsával.

**FONTOS** Az átalakítással vagy létrehozással kialakított dokumentum nem sablonon alapul, a munkálatok olyan Worddel is folytathatók, amelyekre nincs telepítve a bővítmény.

## FELADATOK KIALAKÍTÁSA

Az egyszerűbb feladatok utasításból, adatbekérőből és pontszámból állnak. Ha a pontszámot nem adjuk meg, akkor az 1 lesz.

▪ Döntse el az alábbi állításokról, hogy azok igazak vagy hamisak! ¶

- ✓ → A-feszültség-potenciálkülönbség. ¶
- ✗ → A-feszültség-csak pozitív-lehet. ¶
- ✗ → A-feszültség-iránya az alacsonyabb-potenciál-felől mutat a magasabb felé. ¶
- ✗ → A-feszültségmérő-műszert a terheléssel sorba kötjük. ¶
- ✓ → A-feszültség-a-W-munka-és-a-Q-töltés-hányadosa. ¶

1-pont ¶

Az ábrán látható első bekezdés *Utasítás* stílusú (ALT+U), egy állítás lehet *Állítás (igaz)* vagy *Állítás (hamis)* stílusú (ALT+I, illetve ALT+H), a ✓ vagy ✗ szimbólum alapján ismerhető fel, a kék betűszínű pontszám *Pontszám* stílusú (ALT+P). A pont szó megadása nem kötelező, a pontszámot balról az első számjegysorozat adja.

## STÍLUSOK

A stílusok 5 fő csoportra vannak felosztva. Az elsőbe azok tartoznak, amelyek nem kerülnek az eredménybe vagy használatuk nem kötelező, mert betűformázással is helyettesíthetők. Az alapstílusok többségét használjuk a feladatok elkészítéséhez, ilyen például az *Utasítás* és a *Pontszám*. Az adatbekérők és a feladateleírók az azonos nevű elemek létrehozására használhatók. Az ötödik csoportba egy stílus tartozik, amivel közvetlen kód megadást végezhetünk.

A felhasználók a feladatok begépelésekor általában nem módosítják a bekezdések stílusát, egy ilyen dokumentum bekezdései a stílusok elkészítése után többnyire ibolyaszínűek lesznek. A szín alapján látjuk, hogy mely szövegrészeket nem formáztuk még meg. A *Normál*

stílusú bekezdések nem kerülnek az XML-fájlba, ezért olyan adatok megadására használhatók, amelyek a későbbiekben fontosak lehetnek a feladatok beazonosításában vagy az utómunkálatok elvégzésénél.

## ALAPSTÍLUSOK

**Megjegyzés** • A bekezdés szövege egy megjegyzés csomópont lesz az XML-fájlban.

**Utasítás** • A feladatok utasításainak bekezdésstílusa. Alap esetben ezzel kezdődik a feladat.

**Pontszám** • Bekezdés- és karakterstílus a pontszámok megadásához. A külön bekezdésben megadott pontszám az utasítás és a bekezdés közötti adatbekérőkre vonatkozik, beállítása adott esetben bonyolultabb szabályok alapján is történhet. Ha részpontozást kérünk egy blokk szintű elemre, helyezzük el a *mérleg* vagy *arányos* szövegrészeket a bekezdésben, pl.:

3 pont, mérlegpontozás

4 pont, arányosan

A sorközi elemekre a pontszám arányosan oszlik szét. Amennyiben az egy elemre eső pont-rész nem egész, a bekérőket a program összecepatolja, és a pontszámot az előre állítja be. Táblázatoknál a soronként történő pontozás kétféle módon állítható be. A *soronként 2 pont* azt jelenti, hogy minden sor 2 pontot ér, a *soronként pontozva 6 pont* viszont 2 sor esetén soronként 3 pontot jelent.

A sorközi bekérők után karakterformázással megadott pontszám az adott bekérőre vonatkozik. Ha sorközi pontbeállítás tartalmazza a *pont* szót, a pontjelzés benne marad a feladat szövegében. Az alábbi két mintából a második törlődik a bekezdésből.

(3 pont)

(4)

Csatoláshoz a pontmegadás nem tartalmazhat számjegyet, de legyen benne *cs* szövegrész.

csatolt

(cs)

A pontszámok beállítását tartalomvezérlőkkel is elvégezhetjük. A sorközi bekérőknél megadott értéket a program a pontszétoztásnál figyelembe veszi. Ha például három számadathoz összesen 7 pontot állítunk be, és az egyik vezérlőjében 3 pontot írunk elő, akkor a másik kettő mező 2-2 pontos lesz.

**Bekezdés** • A feladatléíró és az adatbekérőket tartalmazó bekezdések stílusa.

**Táblázat** • A táblázatok celláinak stílusa. Az azonos háttérszínnel megformázott sorok címsorok lesznek.

## ADATBEKÉRŐK

Az adatbekérők felépítése több esetben összetettebb annál, minthogy minden részüket külön stílussal jelöljük, ezért sokszor az egyszerűbb formázást néhány szabállyal egészítjük ki.

**Adat** • Szöveg, szám vagy dátum bekéréséhez használható betűstílus. Az alkalmazott elemet a program az adat típusa alapján választja ki. Ha az automatikus felismerés nem felel meg, például kettes számrendszerű számot szöveggént kell bekérnünk, akkor a kívánt elemtípus vezérlővel állítható be.

<feladatblokk>¶

Egy bolt ötféle-egykilós-kiszerelesű-lisztet árul. Az egyes-kiszerelesek ára és az-elmúlt-hónap-forgalma a-következő-volt:¶

Termék-megnevezése¶	Termék-ára-(forint)¶	Eladott-mennyiség-(darab)¶
rétesliszt¶	269¶	58¶
Graham-búzaliszt¶	239¶	42¶
durum-tésztaliszt¶	369¶	16¶
búzafinomliszt¶	229¶	114¶
pizzaliszt¶	254¶	27¶

▪ Számítsa ki az alábbi adatokat!¶

A lisztek-árának-súlyozatlan-számtani-átlaga:  $272,00 \pm 3\%$  (5-pont)¶

A lisztek-árának-súlyozatlan-szórása:  $50,36 \pm 3\%$  (5-pont)¶

A lisztek-árának-eladott-mennyiséggel-súlyozott-számtani-átlaga:  $251,00 \pm 3\%$  (6-pont)¶

A lisztek-árának-eladott-mennyiséggel-súlyozott-szórása:  $34,34 \pm 3\%$  (6-pont)¶

A pizzalisztból-származó-árbevétel-a lisztek-árbevételének:  $10,63 \pm 3\%$  százalékát-teszi-ki-a. (2-pont)¶

A pizzalisztból-eladott-mennyiség-a lisztek-mennyiségének:  $10,51 \pm 3\%$  százalékát-teszi-ki. (2-pont)¶

A módusz: 229 (3-pont)¶

Az-aszimmetria-mérőszáma (A):  $0,64 \pm 3\%$  (3-pont)¶

</feladatblokk>¶

A fenti példában a törteket két tizedesjegy pontossággal kérjük be, a 3%-os relatív tűrést a  $\pm$  jel után adtuk meg. A táblázat celláinak stílusa *Táblázat* (ALT+T), az utasításon kívül a bekezdések *Bekezdés* stílusúak (ALT+S). A pontszámokat adatbekérőnként adtuk meg úgy, hogy azok szövege megjelenjen a feladatlapon.

**Állítás (igaz), Állítás (hamis)** • Igaz-hamis feladatok állításainak stílusai.

**Válasz (jó), Válasz (rossz)** • A feleletválasztások válaszlehetőségeinek stílusai. A *Válasz (jó)* stílusú bekezdések (ALT+J) előtt pipát tartalmazó négyzet jelenik meg, ezeket kell megjelölni. A *Válasz (rossz)* stílusú bekezdések (ALT+R) az üres négyzetről ismerhetők fel.

Az alábbi mintában 4 válaszlehetőséget adtunk meg. Az alapértelmezett beállítások mellett generáláskor a válaszlehetőségek véletlenszerűen keverednek.



▪ Az alábbiak közül melyik személyiségi jog?¶

- A névviseléshez való jog.¶
- A tulajdonjog.¶
- Az örökléshez való jog.¶
- A szerződési szabadság.¶

1-pont¶

A pontszámot az utolsó válasz után kell feltüntetni, amit jelen esetben el is hagyhattunk volna, mivel az 1. A bekérő további jellemzői tartalomvezérlővel állíthatók be.

**Forrás, Nyelv** • A fogdésvidd bekérő építőelemeinek stílusai. Az alábbi mondatkiegészítésnél a *Bekezdés* stílusú szöveg bézs háttérszínű *Nyelő* karakterstílussal formázott (ALT+N) szavait kell pótolni. Az utasítás alatti *Forrás* stílusú (ALT+F) bekezdésben további, felnem használandó szavakat soroltunk fel. A szavak felsorolásához vessző, perjel vagy virgula használható, a program ezek közül a legelőször előfordulót tekinti elválasztójelnek. A nyelőkben az elemek felsorolási szabálya ugyanez. A nyelőkben felsorolt elemek szerepelhetnek a forráselemek között is, a program az ismétlődő elemekből csak egyet tart meg.

▪ Pótolja a hiányzó szavakat!¶

név, forgó, lehetséges, kibocsátási, ügynöki, tartós, mérleg, pénz, tőzsdei, tartós¶

A befektetések, értékpapírok **bekerülés** értékét úgy állapítjuk meg, hogy az érték fizetett ellenértékből indulunk ki, aminek **befektetett** eszközök esetén mindig részét képezi a **bizományosi** díj és a vásárolt **vétel** opció díja. A befektetett **kamatozó** értékpapírok esetén viszont nem lehet része a beszerzési árnak az abban megfizetett kamat összege.¶

5-pont, arányos¶

A forrásbekezdés megadása nem kötelező. Az alábbi példában olyan elemek is találhatóak, amelyeknek része a perjel. Mivel ez elválasztó karakter, két helyen virgulát helyeztünk el a nyelő elején. Ebben az esetben a például a férfiaság/nőiesség egy elem lesz.

▪ Párosítsa a Hofstede által leírt kulturális dimenziókat a meghatározásaikkal!¶

**hatalmi távolság** – Azt jelzi, hogy az adott kultúrában milyen mértékben fogadják el a hatalom egyenlőtlen eloszlását, hogyan viszonyulnak a hierarchiához.¶

**bizonytalanság-kerülés** – Azt mutatja meg, hogy az emberek, hogyan tolerálják az ismeretlent, a váratlan helyzeteket és a kockázatot.¶

**kollektívizmus/individualizmus** – Azt tükrözi, hogy az emberek inkább a csoportot, a csoportmunkát, vagy az egyént, az egyéni kezdeményezést részesítik előnyben.¶

**férfiasság/nőiesség** – Azt tükrözi, hogy az adott kultúrában melyek a domináns és preferált értékek: a teljesítmény, a kihívások vagy a jó minőségű kapcsolatok és a másokkal való törődés.¶

**időorientáció** – Azt mutatja meg, hogy a társadalom tagjait inkább a rövid vagy a hosszabb távú gondolkodás jellemzi.¶

arányosan 5-pont¶

Az üres nyelőt egy elválasztó karakter megadásával is jelölhetjük, de a / / vagy | | is elfogadott, ami a szövegben jobban látható.

**Rendezés** • A rendezendő elemek bekezdésstílusa. Az elemek értékét egész számként a bekezdés elején tabulátor előtt adjuk meg. A helyes megoldáshoz az elemeket értékük alapján kell majd nemcsökkenő sorrendbe tenni.

Rakja a műveleteket a végrehajtásuk szerinti sorrendbe! Kezdje a legmagasabb prioritásúval!

- 1 → hatványozás (^)
- 2 → egész osztás/hányadosa (÷)
- 3 → összeadás és kivonás (+, -)
- 4 → kisebb vagy egyenlő (<=)
- 4 → Like
- 5 → logikai és (And)

2 pont

A dokumentumban a rendezendő elemeket tetszőleges sorrendben adhatjuk meg, de kisebb a hibalehetőség, ha a feladat készítésekor a jó megoldás szerint látjuk azokat. A feladatlapon a program mindig összekeveri az elemeket.

**Relációanalízis** • A relációanalízis összetett mondatatinak bekezdésstílusa. A tagmondatok igazságértékét és a köztük lévő összefüggést a bekezdés elején tabulátor előtt kell megadni.

Döntse el az alábbi állításról, hogy a tagmondatok tartalma igaz-e, illetve van-e összefüggés a két tagmondat között!

iiy → A CAPM szerint a piac a részvény egyedi kockázatáért nem fizet kockázati prémiumot, mivel az egyedi kockázat diverzifikációval kiküszöbölhető.

2 pont

A megoldásban az első betű az első tagmondatra vonatkozik, a második a másodikra, értékük  $i$  vagy  $h$  lehet. A harmadik betű  $v$  vagy  $n$ , a tagmondatok közötti összefüggésre utal.

**Lenyíló lista** • A lenyíló vagy léptető lista karakterstílusa (ALT+L), amely a szövegben szürke háttérrel és szaggatott szegéllyel jelenik meg. Az elemek felsorolásához vessző, perjel vagy virgula használható, a program ezek közül a legelőször előfordulót tekinti elválasztójelnek. A helyes megoldásokat \*-gal jelezhetjük az elem elején. Amennyiben nincs csillaggal megjelölt elem, az esőként megadott lesz helyesnek beállítva.

Pótolja a hiányzó szavakat!

A kettős könyvvizetés teljesítés / profit / cash-flow (vagy eredmény) szemléleten alapul, azaz figyelembe vesz minden vagyoni és eredményváltozást (gazdasági eseményt) függetlenül attól, hogy pénzügyi / az adott időszakban történik-e vagy sem.

2 pont

Az ábrán látható mintában az első listának 3, a másodiknak 2 eleme lesz, mindkettőnél az első elem a helyes megoldás.

**Jelölő** • A jelölők karakterstílusa, amely a szövegben szürke háttérrel és pontozott szegéllyel jelenik meg. Ha a *Jelölő* stílussal megformázott szöveg  $*$ ,  $x$  vagy  $X$ , akkor jelölőnégyzetbe pipát kell tenni a helyes megoldáshoz.

▪ Jelölje a gazdasági eseménynél, hogy melyik vagyონrész hogyan változott!¶

Gazdasági esemény¶	Eszköz¶		Forrás¶		Költség¶		Ráfordítás¶		Bevétel¶		Eredmény¶	
	+¶	-¶	+¶	-¶	+¶	-¶	+¶	-¶	+¶	-¶	+¶	-¶
Fizetett kártérítés¶	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2-pont-soronként¶

A fenti mintában a  $*$ -ot tartalmazó cellákba kell majd pipát tenni. A megoldás 2 pontot ér, ha pontosan ezt a 3 jelölőnégyzet jelöljük be.

## FELADATLEÍRÓK

A feladatok leírásakor legtöbbször bekezdést, táblázatot, illetve ábrát használunk. Az első két elemről korábban már volt szó, az ábrák konvertálása automatikusan történik. A többi feladatleíró elem ritkábban használ, azok stílusai alapesetben rejtve vannak, ha szükség van valamelyikre azt, jelenítsük meg a **Stílusok** panelen.

**Cím** • Ezzel a bekezdésstílussal címsort hozhatunk létre a feladatlapon.

**Felsorolás** • A díszpontos felsorolás bekezdéseinek stílusa.

**Számozott lista** • Számozott vagy betűzött listák elemeinek bekezdésstílusa. Alapesetben a számozás arab számokkal történik, másfajta számozás tartalomvezérlővel állítható be.

**Forráskód** • A forrásprogramok és kódrészletek bekezdés- és karakterstílusa.

**Képlet** • LaTeX vagy AsciiMath formátumban megadott képletek karakterstílusa.

**Szószedet** • Karakterstílus szószedet készítéséhez. A megformázott szövegrészben tegyünk egyenlőségjelet a megmagyarázott kifejezés és a buborékban megjelenő szöveg közé.

□ → explicit=kifejtett, nyelvi kifejezéssel jelölt típusátalakítás¶

**Letöltés** • Karakterstílus, amellyel letölthető fájlokat adhatunk a feladatlaphoz. A forrásfájl specifikációját szögletes zárójelben kell megadni. Az alábbi mintán az egyenlőségjel után az *alias* jellemző értéke szerepel, a mentés ezzel az azonosítóval történik.

Töltse le a `fgell.p[matlab/fgell01.p=fgell.p]` függvényfájlt, majd hívja meg a `fgell` függvényt az Ön által készített függvénnyel mint paraméterrel:¶

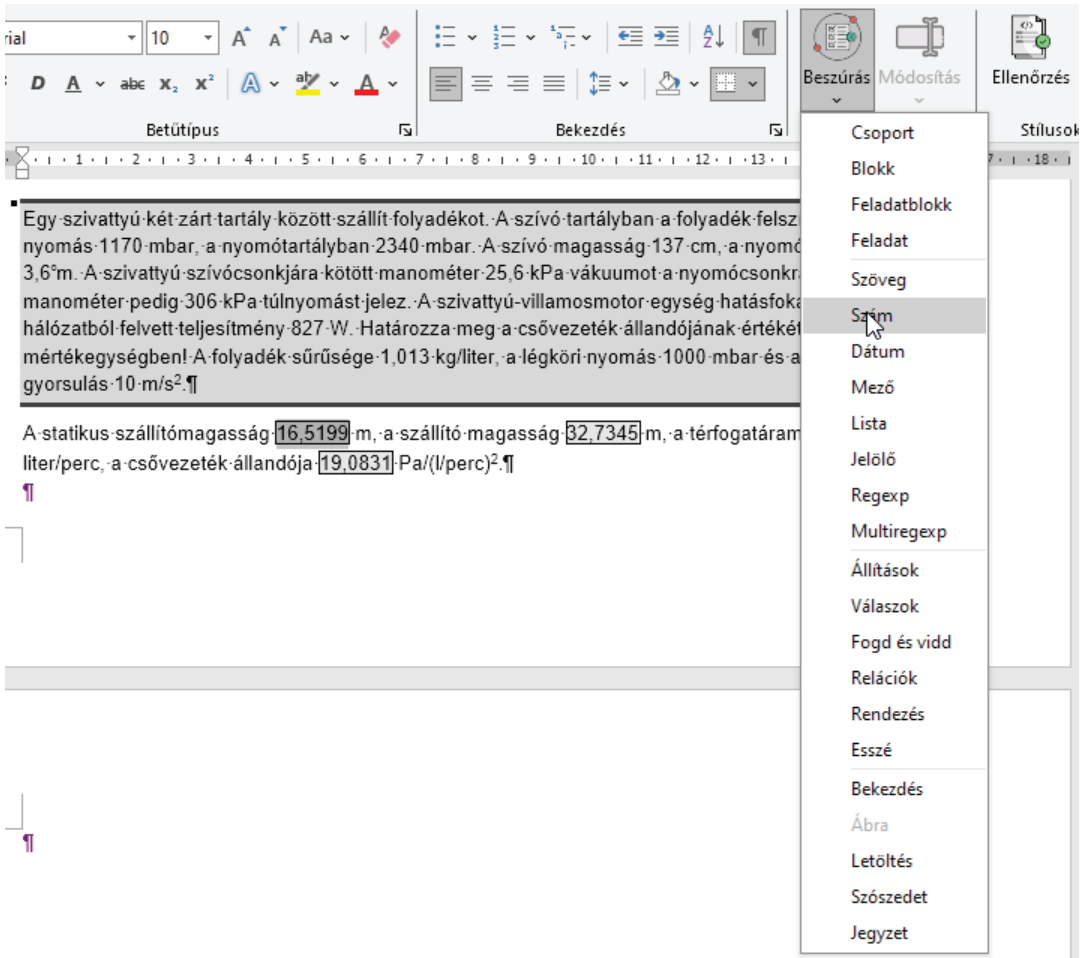
## KÖZVETLEN KÓDMEGADÁS

Az *XML-kód* bekezdésstílussal (ALT+X) XML-kódrészleteket helyezhetünk a dokumentumba. Elsősorban csoportok, blokkok és feladatblokkok készítésére használjuk. A bekezdések

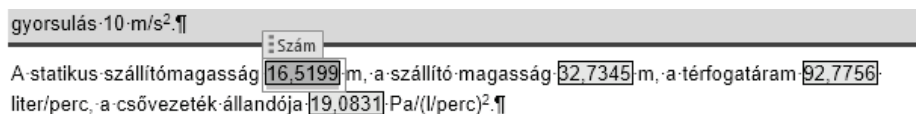
a sárga háttérszínről ismerhetők fel, szintaktikai és szemantikai helyességükről nekünk kell gondoskodnunk. Az elhelyezett XML-kód ellenőrzését a **Stílusok** csoport **Ellenőrzés** parancsával végezhetjük el.

## TARTALOMVEZÉRLŐK

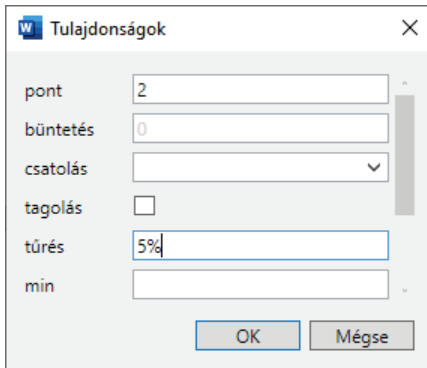
A dokumentum megformázásával létrehozhatjuk az elemeket, azonban ahhoz, hogy a jellemzők többségét beállíthassuk, tartalomvezérlőket kell elhelyeznünk. Vezérlő beszúrásához jelöljük ki az adott elem szövegrészét, majd a **Vezérlők** csoport Beszúrás menüjéből válasszuk ki a megfelelő elemet!



A kijelölt szövegrész körül láthatóvá válik a kiválasztott menüvel megegyező nevű tartalom-vezérlő.



Az elem jellemzőit a megjelenő **Tulajdonságok** párbeszédpanelben állíthatjuk be. Az ablak mezőiben az alapértelmezett értékek szürke betűszínűek.



A beállítások elvégzését követően a módosított jellemzők neve és értéke zárójelben jelenik meg a vezérlő neve után.

gyorsulás: 10·m/s<sup>2</sup>.¶  
 Szám (pont=2; tűrés=5%)  
 A statikus szállítómagasság 16,5199 m, a szállító magasság 32,7345 m, a térfogatáram 92,7756 liter/perc, a csővezeték állandója 19,0831 Pa/(l/perc)<sup>2</sup>.¶

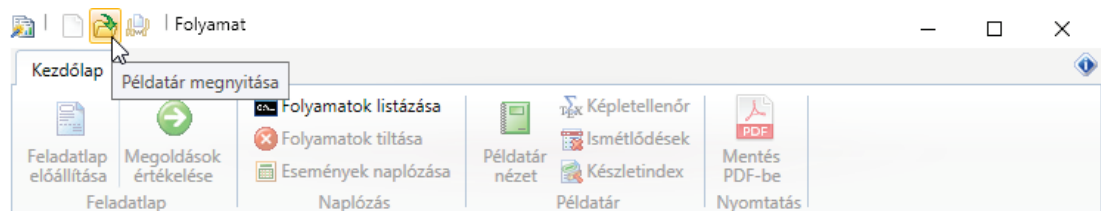
A Szöveg, Szám, Dátum, Mező, Regexp és Multiregexp vezérlők a **Módosítás** menüvel másik típusra változtathatók. Ha tévesen szűrtünk be egy vezérlőt vagy nincs rá szükségünk, a helyi menü **Tartalomvezérlő eltávolítása** parancsával szüntethetjük meg.

## PÉLDATÁRAK TESZTELÉSE

Az elkészített feladatokat a Folyamat segítségével ellenőrizhetjük. A program neve arra utal, hogy a beszámolóval kapcsolatos műveletek mindegyike sorra végrehajtható: feladatlap előállítás, megjelenítése, alkalmazások tiltása és naplózása, megoldások gépi értékelése a javítókulcs alapján, átpontozás.

A példatár megnyitásához hajtsuk végre az alábbi műveletek valamelyikét:

- Nyomjuk meg a **Példatár megnyitása** gombot a gyorselérési eszköztáron, majd a megjelenő párbeszédablakban válasszuk ki az XML-fájlt!
- Fájlkézelőből húzzuk a fájlbejegyzést a programablakra!



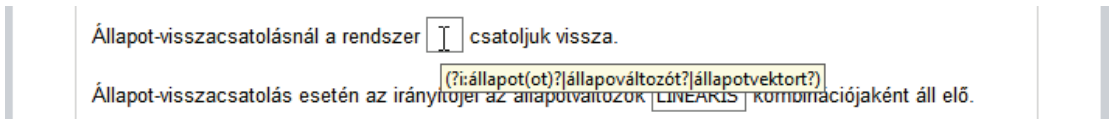
Indításkor a program automatikusan megnyitja az utoljára használt fájlt, ha az még létezik. A megnyitott fájl specifikációja a címsorban látható.

Az újonnan készült példatáraknál célszerű először a **Példatár nézet** parancssal megjeleníteni az összes feladatot. Nézzük végig a tartalmat, és ellenőrizzük le, hogy minden megfelelően jelenik-e meg. Az XML-fájl módosításakor a képernyő a parancs újbóli kiválasztásával vagy az F5 funkcióbillentyű megnyomásával frissíthető.

The screenshot shows the IMPERA software interface. At the top, the window title is 'Folyamat - F:\Munka\Impera\vizsga.xml'. Below the title bar is a 'Kezdőlap' (Start page) menu with several icons and labels: 'Feladatlap előállítása', 'Megoldások értékelése', 'Folyamatok listázása', 'Folyamatok tiltása', 'Események naplózása', 'Naplózás', 'Példatár nézet', 'Képletellenőr', 'Ismétlődések', 'Készletindex', 'Példatár', and 'Mentés PDF-be', 'Nyomtatás'. The main content area displays the title 'Stratégiai menedzsment' in green. Below the title, the subject code 'KGLB\_MMTM044', the exam type 'vizsga', and the duration '50 perc' are listed. The test is divided into three sections: '1. feladat', '2. feladat', and '3. feladat', each containing a 'Csoport' (Group) box. The first section contains a question: 'Döntse el az alábbi állításról, hogy a tagmondatok tartalma igaz-e, s van-e összefüggés a két tagmondat között! Jelölje be a megfelelő válasz betűjelét!'. The options are: A: Az első tagmondat igaz, a második is igaz, továbbá ok és okozati összefüggés van köztük. B: Az első tagmondat igaz, a második is igaz, de ok és okozati összefüggés nincs köztük. C: Az első tagmondat igaz, a második hamis. D: Az első tagmondat hamis, a második igaz. E: Mindkét tagmondat önmagában is hamis. Below the question, a paragraph of text is shown: 'A pénzügyi irányítási stílusban az SBU-k magas fokú önállóságot élveznek, mert a vállalati központ csak a pénzügyi célok monitoringját végzi.' To the right of the text are five radio buttons labeled A, B, C, D, E, with the first one (A) selected. The question is worth 2 points. The second section contains a similar question.








**FONTOS** A példatár megjelenítése, illetve feladatlap előállítás csak akkor lehetséges, ha az XML-fájl érvényes a *peldatar.xsd* sablon alapján.

Példatár nézetben az adatbekérők megoldásai többnyire közvetlenül leolvashatók. Reguláris kifejezéssel történő ellenőrzésnél a mezőbe az elemnév kerül szürke színnel, a megoldás jellemző értéke, vagy ennek hiányában a reguláris kifejezés buborékban jelenik meg. Ugyancsak buborékban láthatók a szöveg elem szinonimái.



**MEGJEGYZÉS** Az önépítő feladatoknál minden elem megjelenik, a sorsolás módja csak az XML-fájlból olvasható ki.

Néhány jellemző értékét ikon jelöli, ezáltal láthatjuk a fontosabb beállításokat:

-  csatolás az előző bekérőhöz
-  pontszétosztás, vagyis az összecsatolt adatbekérőkre szétosztjuk a megadott pontszám
-  önépítő (generatív) feladat
-  részpontozás
-  mérlegpontozás
-  állandó sorrend
-  újrakevert

Hiba esetén a csoport és az azon belüli elemszám alapján kereshetjük meg a feladatot. Bonyolult szerkezetű példatáraknál húzzuk az egeret az adott feladat fölé, és a kis idő múlva megjelenő buborékból olvassuk le a példatárbeli sorszámát, amely alapján egy XML-szerkesztőben gyorsan eljuthatunk a hiba helyéhez. Az Oxygenben például a 47-es számú feladat az alábbi XPath-kifejezéssel található meg a legegyszerűbben:



**MEGJEGYZÉS** A csoportot jelölő sárga háttérű téglalapban a leírás jellemző értéke, ennek hiányában a *Csoport* felirat jelenik meg. Ha csoportok leírását megfelelően választjuk meg, könnyebben megtalálhatjuk a keresett feladatot.

## FELADATLAPOK VIZSGÁLATA

A **Feladatlap előállítás** parancs kiválasztásakor a program megjeleníti a példatár alapján elkészített feladatlapot. A nyomógomb újbóli megnyomásával vagy az F5 funkcióbillentyűvel új feladatlap generálható.

A kitöltött feladatlap a **Megoldások értékelése** paranccsal lepontozható. Az **Adatok** lapon az elért és az összes elérhető pontszám mellett adatbekérőnként láthatjuk a javítókulcsokat és a kigyűjtött adatokat.

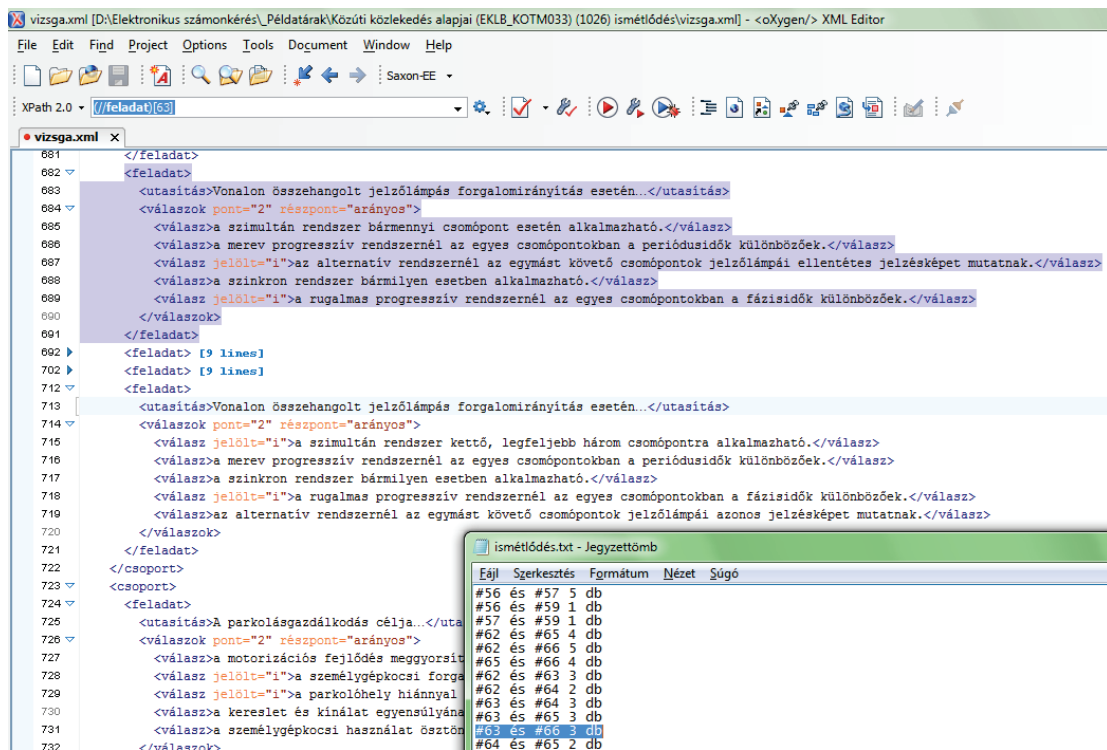
Javítókulcs	Megoldás: 10/100 (10%)
<pre>&lt;kulcsok&gt; &lt;kulcs elem="válaszok" pont="10" büntetés="0" részpont="nincs"&gt;innr &lt;kulcs elem="válaszok" pont="10" büntetés="0" részpont="nincs"&gt;nnn &lt;kulcs elem="válaszok" pont="10" büntetés="0" részpont="nincs"&gt;innr &lt;kulcs elem="válaszok" pont="10" büntetés="0" részpont="nincs"&gt;innr &lt;kulcs elem="válaszok" pont="10" büntetés="0" részpont="nincs"&gt;nnir &lt;kulcs elem="állítások" pont="10" büntetés="0" részpont="arányos"&gt;iiii &lt;kulcs elem="állítások" pont="10" büntetés="0" részpont="arányos"&gt;hi &lt;kulcs elem="állítások" pont="10" büntetés="0" részpont="arányos"&gt;ih &lt;kulcs elem="szöveg" pont="2" büntetés="0" javítás="gépi" betűérzéke &lt;kulcs elem="szöveg" pont="2" büntetés="0" javítás="gépi" betűérzéke</pre>	<pre>&lt;megoldások&gt; &lt;adat pont="10"&gt;innnn&lt;/adat&gt; &lt;adat pont="0"&gt;nnnnn&lt;/adat&gt; &lt;adat pont="0"&gt;nnnnn&lt;/adat&gt; &lt;adat pont="0"&gt;nnnnn&lt;/adat&gt; &lt;adat pont="0"&gt;nnnnn&lt;/adat&gt; &lt;adat pont="0"&gt;nnnnn&lt;/adat&gt; &lt;adat pont="0"&gt;00000&lt;/adat&gt; &lt;adat pont="0"&gt;00000&lt;/adat&gt; &lt;adat pont="0"&gt;00000&lt;/adat&gt; &lt;adat pont="0"&gt;00000&lt;/adat&gt; &lt;adat pont="0"&gt;&lt;/adat&gt; &lt;adat pont="0"&gt;&lt;/adat&gt;</pre>

## KÉPLETEK ELLENŐRZÉSE

A **Képletellenőr** parancs kiválasztásakor a program a dokumentum LaTeX-képleteit sorra bejárva a { } zárójelpárokat ellenőrzi. Az első előforduló hibánál a képlet a vágólapra kerül, ami alapján az megkereshető a fájlban.

## ISMÉTLŐDÉSEK KISZÜRÉSE

Az **Ismétlődések** parancs feltérképezi a példatárban többször előforduló válaszokat és állításokat, és ezekről jelentést állít össze, amit a példatár mellé ment az *ismétlődés.txt* fájlba.



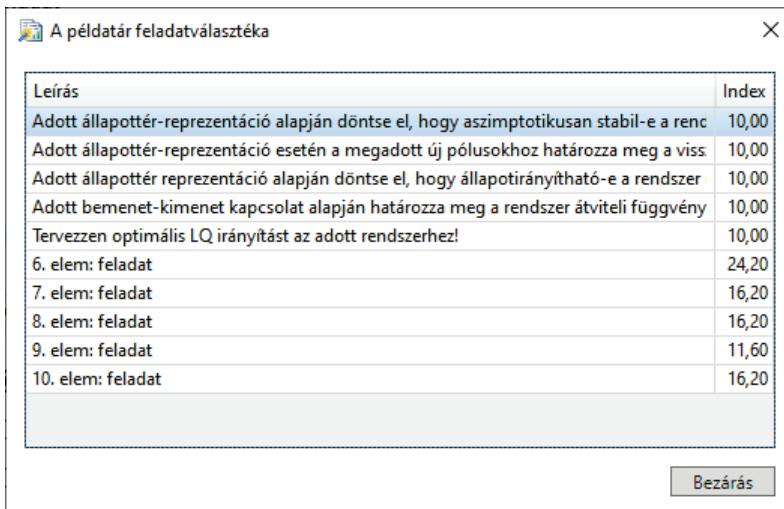


Az adott feladatokat a feladatazonosítók alapján kereshetjük ki. Az összehasonlítás egyszerűbb, ha összecsuksukjuk a környező elemeket, hogy mindkét feladat látható legyen, vagy ha ez nem megoldható, vertikálisan osszuk ketté a szerkesztőt.

**MEGJEGYZÉS** A hasonló feladatokat célszerű egyesíteni. Szüntessük meg az ismétlődő válaszokat vagy állításokat, és csoportok kialakításával biztosítsuk az adott számú elem sorsolását. Az így kialakított önépítő feladatok könnyebben átláthatók, mivel kevesebb kóddal írhatók le, és az esetek többségében többféle változatban kerülhetnek a feladatlagra.

## KÉSZLETINDEX

A példatár feladatválasztékáról a **Készletindex** paranccsal összesítést kaphatunk. A listában a dokumentumelem azon gyerekelemei szerepelnek, amelyek adatbekérőt tartalmaznak.



Leírás	Index
Adott állapotér-reprezentáció alapján döntse el, hogy aszimptotikusan stabil-e a renc	10,00
Adott állapotér-reprezentáció esetén a megadott új pólusokhoz határozza meg a viss	10,00
Adott állapotér-reprezentáció alapján döntse el, hogy állapotirányítható-e a rendszer	10,00
Adott bemenet-kimenet kapcsolat alapján határozza meg a rendszer átviteli függvény	10,00
Tervezzen optimális LQ irányítást az adott rendszerhez!	10,00
6. elem: feladat	24,20
7. elem: feladat	16,20
8. elem: feladat	16,20
9. elem: feladat	11,60
10. elem: feladat	16,20

Bezárás

Az ábrán egy olyan összeállítás mutatói láthatók, amelynél a feladatlagra 10 feladat kerül. Az első 5 sor olyan csoportokhoz tartozik, amelyekben 10-10 egyszerű feleletválasztós feladat van. Az utolsó 5 sor mindegyikéhez önépítő feladat tartozik, ezeknél nincs megadva a leírás jellemző. Az indexeknél a tört szám annak köszönhető, hogy például a 7. elemnél 81 mondat közül választjuk ki az 5-öt.