

## 47. mintafeladat – ismerkedés a listákkal

1. Álljunk a parancsértelmező ablakba (**Python 3.8.0 Shell**).
2. A prompt jel >>> mögött villog a kurzor. Oda írjuk majd be a parancsokat, mindegyik után megnyomva az **Enter** billentyűt.
3. Készítsünk az első három hónap nevéből egy *honap* nevű listát. A listát szögletes zárójelek közé írjuk, elemeit vesszővel választjuk el egymástól. Azért vannak időzőjelek is, mert karakterláncok az elemek. Írjuk be, majd üssünk **Enter**:  
`honap = ["január", "február", "március"]`
4. Visszakaptuk a prompt jelet, jöhet az újabb parancs. Írassuk ki a lista 2 indexű elemét. Írjuk be, majd üssünk **Enter**:  
`print(honap[2])`
5. Azt írja ki a parancsértelmező, hogy március.
6. Írassuk ki most az egész listát. Írjuk be, majd üssünk **Enter**:  
`print(honap)`
7. Kiíródik az összes elem, zárójelestül, aposztrófokkal, vesszőkkel. Ez így használhatatlan lesz a programban, mert ott általában valamilyen formázott kiírás az elvárás. (Például: január 23.) Viszont nagyon hasznos lehet tesztelési céllal, hogy lássuk, éppen mi van a listában.
8. Hozzunk létre egy *nev* nevű üres listát. Írjuk be, majd üssünk **Enter**:  
`nev = []`
9. Próbáljunk értéket adni a lista 0 indexű elemének. Írjuk be, majd üssünk **Enter**:  
`nev[0] = "Glázser Bozsó"`
10. Hibaüzenetet kapunk: *IndexError: list assignment index out of range*. Ennek az oka, hogy a lista üres, tehát nincs 0. eleme, így minek is adnánk értéket.
11. Listába (így üres listába is) új elemet felvenni az **append** utasítással lehet. Írjuk be, majd üssünk **Enter**:  
`nev.append("Glázser Bozsó")`
12. Most már, hogy van 0 indexű eleme a listának, kiírathatjuk. Írjuk be, majd üssünk **Enter**:  
`print(nev[0])`
13. Megváltoztathatjuk a listaelemet értékadással. Írjuk be, majd üssünk **Enter**:  
`nev[0] = "Nyomasek Bobó"`
14. Ellenőrzésként ismét írassuk ki a 0 indexű elemet. Ha minden rendben, Nyomasek Bobót fogjuk kapni. Írjuk be, majd üssünk **Enter**:  
`print(nev[0])`
15. Másképpen is létrehozhatunk egy listát (a példában a lista neve *szuletes*). Ha tudjuk, hogy kezdetben 40 elemű lesz, akkor feltöltjük létrehozáskor 40 db [0] elemmel. Írjuk be, majd üssünk **Enter**:  
`szuletes = [0]*40`

16. Adjunk értéket a 29 indexű elemnek. Legyen az értéke 1. Írjuk be, majd üssünk **Entert**:  
`szulettes[29] = 1`
17. Ellenőrzésként írassuk ki a 29 indexű elemet. Ha minden rendben, az 1-et fogjuk kapni. Írjuk be, majd üssünk **Entert**:  
`print(szulettes[29])`
18. Nézzük meg, hogyan lehet egy index maga is listaelem. Írjuk be, majd üssünk **Entert**:  
`print(honap[szulettes[29]])`
19. Eredményként a februárt kapjuk. A `szulettes[29]` értéke 1, így a hónap mögötti zárójelbe az 1 kerül. A kiírt elem tehát *február* lesz:  
`honap[szulettes[29]] → honap[1] → február.`

## 48. mintafeladat – néhány listaművelet

Egy játékba tippeket küldtek számok formájában, amiket az érkezés sorrendjében jegyeztek le. Az nyer, aki a legkisebb számra tippel *egyedül*. Próbáljunk ki pár listaműveletet rajta.

1. Álljunk a parancsértelmező ablakba (**Python 3.8.0 Shell**).
2. A prompt jel `>>>` mögött villog a kurzor. Oda írjuk majd be a parancsokat, mindegyik után megnyomva az **Enter** billentyűt.
3. Adjuk meg a következő listát, azaz írjuk be a parancssorba:  
`tippek = [3, 12, 1, 8, 5, 8, 1, 2, 1, 4]`
4. A helyes tipp a 2 volt. Hányadiknak érkezett? Ennek megállapításához használjuk az **index** függvényt, ami az elem sorszámát adja meg. Ehhez még hozzá kell adnunk 1-et, hogy ne a 0-val kezdődjön a számlálás:  
`tippek.index(2) + 1`
5. Készítsünk *masolat* néven a *tippek* tömbről egy valódi másolatot, azaz a *tippek* és *masolat* tömb egymástól függetlenül módosíthatók legyenek. Erre a célra szolgál a **copy** függvény:  
`masolat = tippek.copy()`

6. A 4. és 5. *tippek* beadói csaltak, ezért töröljük őket a *tippek* listából. Indexszel megadott elemeket a `del` paranccsal törölhetünk. A megadott sorszámokhoz képest az index 1-gyel kisebb, így a törlendő tartomány [3:5], hiszen az 5.-et már nem fogjuk törölni:  
`del tippek[3:5]`
7. Ellenőrizzük a maradék listát, tényleg a megfelelő elemeket töröltük-e (igen):  
`tippek`
8. Rendezzük a *tippek*et növekvő sorrendbe, hogy jól látszódjon, melyik a legkisebb, nem ismétlődő szám, mert az a nyerő tipp. Ezt a **sort** függvény segítségével tehetjük meg<sup>42</sup>:  
`tippek.sort()`
9. Ellenőrizzük a listát, megtörtént-e a rendezés (igen):  
`tippek`
10. Nézzük meg a listát csökkenő sorrendbe rendezve. Ezt úgy tudjuk megtenni, hogy létezik a **reverse** függvény, ami megfordítja az elemek sorrendjét. Ez az utasítás nem csökkenő sorrendbe rendez. Most azért lesz mégis a lista csökkenő sorrendben, mert előtte növekvő sorrendben volt.  
`tippek.reverse()`
11. Ellenőrizzük a listát, megtörtént-e a csökkenő rendezés (igen):  
`tippek`
12. Most ellenőrizzük a *masolat* listát. Ha tényleg független a *tippektől*, akkor nem szabad rendezettné lennie, és két elemmel több van benne.  
`masolat`
13. Készítsünk *hivatkozas* néven a *tippek* tömbből egy nem valódi másolatot, azaz a *tippek* és *hivatkozas* tömb egymástól függetlenül ne legyenek módosíthatók.  
`hivatkozas = tippek`
14. Csak 10 alatti számra lehetett tippelni, így töröljük a 12-es számot a *tippek* listából. Erre a célra szolgál a **remove** parancs:  
`tippek.remove(12)`
15. Ellenőrizzük mindkét listát, megtörtént-e a 12 eltávolítása (igen):  
`tippek`  
`hivatkozas`

---

<sup>42</sup> Ékezetes betűket tartalmazó karakterláncokra nem működik megfelelően.