

Lottó Az alábbi feladatban a 2022. évi 5-ös lottó nyerőszámokkal fogunk dolgozni, amelyeket a mellékelt `szamok.csv`-ben találja (`lotto.csv`). A feladat megoldása során az itt található adatokat kell beolvasni és feldolgozni. Az állomány soraiban az adatok tabulátor karakterrel kerülnek elválasztásra, az első adat a hét sorszáma, azt követő 5 érték pedig az adott heti nyerőszámok. A feladatok megoldása során nem feltételezheti az adatok rendezettségét, tehát nem lehet biztos benne, hogy az állományban a hetek sorrendben kerültek rögzítésre, vagy hogy a héthez rendelt nyerőszámok rendezettek lennének!

1. AJAX kéréssel olvassuk be a `data/lotto.csv` tartalmát egyszerű szöveggént és tároljuk el egy segédváltozóban az `index.html` fájlban. Gondoskodjon arról, hogy az AJAX kérés az oldal teljes betöltése után automatikusan kerüljön végrehajtásra, ehhez használja a megfelelő eseménykezelő függvényt!
2. Hivatkozza be az `index.html` fájlban, a `fajlkezeles.js` fájlt és `feldolgozas.js` forrásállományokat!
3. Implementálja a `src/fajlkezeles.js` állományban a `konvertal` függvényt, amely bemenete egy forrás csv szöveg, visszatérési értéke pedig a forrásállományból konvertált objektum lesz. A feladat megvalósítását az alábbi lépések szerint valósítsa meg:
 - a. Csak egyetlen paramétert fogadunk, amely típusa szöveg és a szöveg paraméter nem lehet üres, különben dobjunk egyszerű hibát! A szöveges tartalom szerkezetét külön nem kell ellenőrizni!
 - b. Hozzon létre egy üres objektum típusú segédváltozót!
 - c. Bontsa a bejövő szöveget először sorokra, a sorlezáró karakter: `\r\n`.
 - d. Járja be a sorokat, minden adatot tartalmazó sort bontson mezőkre, a mezőhatároló karakter a tabulátor.
 - e. A nyerőszámokat (a sorban szereplő utolsó 5 szám) készítsen egy numerikus egészeket tartalmazó tömböt.
 - f. Az objektumban készítsen egy új kulcsot a hét sorszáma megfelelő értékkel, az ehhez rendelt érték pedig legyen az előző feladatban elkészített tömb!

A feladat végén olyan objektumot kell kapnunk, amely kulcsai a hetek sorszáma (numerikus értéként), a hozzárendelt értékek pedig az adott heti nyerőszámok! Ezt az objektumot adja vissza a hívás helyére.

4. Lehet, hogy hibát vétettünk és nem rögzítettünk minden hetet! Implementálja az `src/feldolgozas.js` állományban a `mindenHetSzerepele` függvényt, amely paraméterében kapja az előző feladatban elkészített objektum szerkezetével megegyező paramétert, a hívás végén pedig logikai értékkel tér vissza!
 - a. A függvény egyetlen paraméterrel működik, ha a paraméterszám nem megfelelő, térjen vissza `false` értékkel. A paraméterrel kapcsolatos egyéb ellenőrzéseket nem kell végeznie!
 - b. Az objektum kulcsai a hetek sorszáma, így ezeket a kulcsokat gyűjtse ki egy tömbbe!
 - c. Ciklus segítségével ellenőrizze le, hogy 1-től 52-ig minden szám előfordul-e a tömbben.
 - d. Ha valamelyik sorszám nem fordul elő a tömbben térjen vissza hamis értékkel, különben a visszatérési érték legyen logikai igaz.
5. Egy másik hibát is jeleztek, így azt is meg kell vizsgálnunk, hogy adott héten nem került-e véletlenül ugyanaz a szám többször a nyerőszámok közé. Most azonban azt is jelezni kell, hogy mely számokkal mely heteken volt probléma. Implementálja az `src/feldolgoz.js` állományban a `hibasSzamok` függvényt, amely paraméterében kapja az a korábban elkészített objektum szerkezetével megegyező paramétert. A hívás végén olyan objektummal térünk vissza, amelyben a kulcsok a nyerőszámok, a hozzárendelt érték pedig egy olyan tömb, amely azoknak a heteknek a sorszámaát tartalmazza, amelyen a nyerőszám egynél több alkalommal is előfordult!
 - a. A bejövő paraméterszám csakis egy lehet, ha ennél kevesebb vagy több paraméterrel hívjuk meg, akkor dobjunk hibát!
 - b. Készítsen egy objektumot, amelyben a kulcsok 1-től 90-ig a numerikus értékek, és a hozzárendelt érték minden esetben legyen null. A hívás végén ezt az objektumot fogjuk visszaadni.

- c. Gyűjtse ki a kulcsokat a paraméterben kapott objektumból (hét sorszámk) egy tömbbe.
 - d. Az előző részfeladatban létrehozott tömb segítségével járja be az objektumot hetenként! Ha az adott héten egy nyerőszám többször is szerepel, akkor azt gyűjtse egy segéd tömbbe úgy, hogy minden szám csakis egyszer forduljon elő abban! A futás végén járja be a segéd tömböt, és a az objektum megfelelő számkulcsa alá szúrja be a hét sorszámt (ügyeljen arra, hogy kezdetben minden kulcshoz rendelt érték null)
6. Melyik számot húzták ki a legtöbbszer a 2022-es évben? Implementálja az *src/feldolgoz.js* állományban a *legtobbszorKihuzott* függvényt, amely paraméterében kapja az a korábban elkészített objektum szerkezetével megegyező paramétert. A hívás végén térjünk vissza az évben legtöbbet kihúzott számmal!
- a. A függvény kizárólag egy paraméterrel hívható, ha nem egy paraméterrel hívjuk meg a függvényt, akkor dobjunk fel kivételt!
 - b. Készítsünk egy segéd objektumot, amelyben a kulcsok legyenek a számok 1-től 90-ig, a kulcshoz rendelt érték legyen 0 minden esetben!
 - c. Járjuk be hetenként a nyerőszámokat és dolgozzuk fel az adott heti nyerőszámokat! Az egyes számok feldolgozása során növeljük a korábban létrehozott objektum megfelelő kulcsához rendelt értéket 1-gyel!
 - d. A feldolgozás után válasszuk ki a legnagyobb értékhez tartozó kulcsot és adjuk vissza a hívás helyére. Ha több szám is maximum alkalommal került kihúzásra, akkor adjuk vissza a legnagyobb számot az előfordulók közül!
7. Helyezzünk fel egy gombot az index.html-ben „6. feladat” felirattal, gombra kattintás esetén hívjuk meg a *legtobbszorKihuzott* függvényt. A hívás eredményt jelenítse meg popup ablakban!