

Vásárlás – JS DOM (gyakorló)

Egy vásárlással kapcsolatos weblapot kell készítened! A munkád során kizárólag a **script.js** fájlban végezhetsz módosításokat!



A szkript kezdetén egy `main` nevű függvény fog lefutni. Ebben található egy termékek nevű üres lista, valamint egy képek nevű lista, amely a `pictures` mappában található `.jpg` képek neveit tartalmazza (kiterjesztés nélkül). A `main` 3 fontosabb részt előállító alprogramot hív meg, a `bevasarlo` eljárás pedig további alprogramokat.

Az említett két alprogramot (`main`, `bevasarlo`) ne módosítsd, a többi létrehozása közben viszont további függvényeket szükség szerint definiálhatsz!

A függvényhívások közül nyugodtan helyezd kommentbe azokat, amiket egyelőre nem definiáltál, hogy a weboldal készítése közben folyamatosan tudj tesztelni. (Később ne felejtse őket kivenni kommentből.)

Statisztikák

1. Írj `beolvas` néven eljárást, amely az üres termékek listát feltölti a weboldalon látható bevásárló lista termékeivel és áraival.

A termékek lista elemei termék objektumok legyenek. Minden terméknek legyen neve és ára (előbbi szöveg, utóbbi szám típus)!

Mj.: a lista elemek darabolásához használd fel a `.split` string metódust, amely működésében hasonló a Python és a C# nyelvekben megismert azonos nevű függvénnyel!

2. Készíts `osszesAr` néven eljárást, amely kiszámolja a termékek összesített árát, majd beírja az értéket a számozott listában található megfelelő `span` elembe!

3. A `kerekAr` alprogram megnézi, hogy található-e olyan termék, amelynek ára 0-ra végződik. Amennyiben igen, akkor az „Igen”, máskülönben a „Nem” szót jeleníti meg a listaelem végén egy új `span` elemként!

Vigyázat! A függvénynek egymás után többször futtatva is helyesen kell működnie!

4. Csinálj `dragak` néven eljárást, amely a 2000-nél drágább termékek árát tartalmazó `span` elemekre alkalmazza a `draga` osztályt, továbbá a teljes hozzájuk tartozó lista elemet félkövéren formázza!

Mj.: Ne legyen paramétere az eljárásnak!

5. A `legdragabb` eljárás meghatározza a legdrágább termék nevét, majd beírja a statisztikai részeket tartalmazó harmadik lista elembe található `span`-be!

Új elemek

6. Az `ujTermek` eljárás két paramétert kapjon: egy nevet és egy árat. A kapott értékekből egy új terméket készít, amelyet új elemként rögzít a weboldal tetején látható számozatlan felsorolásban is!

Vigyázat! Az új termék beszállása után a statisztikákat újra kell számolni!

Mj.: Az `ujTermek` eljárás a programon belül sehol sem kerül meghívásra. Teszteléséhez futtasd a konzolról két általad választott paraméterrel. Pl.: `ujTermek („Fagyi”, 500)` vagy `ujTermek („Tarja”, 2979)`.

7. Készítsd el a `kepekBeszuras` alprogramot, amely beszurja a `pictures` mappában található képeket a weboldalon található `kepek` azonosítójú keretbe! A képek magassága egységesen `100px` legyen, helyettesítő szöveggént pedig a kép neve jelenjen meg!

A képek egy `flexbox`-ban helyezkedjenek el úgy, hogy a közöttük lévő távolság egyforma legyen! A szükséges stílus beállításokat is JavaScript segítségével végezd el!

8. A `diagram` eljárásban készíts egy sávdigramot a termékek áráiról!

- a) A sávok `div` elemek legyenek, amelyek szöveggént tartalmazzák a termék nevét! A sávok szélességeit `3000`-hez viszonyítva határozd meg! (Ahány százaléka egy termék ára a viszonyítási alapnak, annyi százalék legyen az elem szélessége!)

Állítsd be azt is, hogy a sávok margója függőlegesen `1em`, háttérszínük világoszöld legyen!

Vigyázat! A `diagram` eljárásnak akkor is helyesen kell működnie, ha egymás után többször hívjuk meg!

- b) Szintén a `diagram` alprogramon belül hozz létre a sávokat követően egy további `div` elemet, amelynek tartalma egy `hr` lesz, melyet két `span` elem vesz körbe a minta szerint!

Ennek a `div` elemnek az azonosítója `felirat` legyen!

9. Tekintsük a `body` azon `div` gyermekeit, amelyeknek van azonosítója! A hivatkozások eljárás készít egy menüt a weboldal tetejére, amelyben az előzőleg említett `div` elemekhez tartozik egy-egy oldalon belüli link.

A linkek szövegei a `div` elemek azonosítói legyenek, de a felhasználó számára mindegyik nagybetűvel kezdődően látszódjon!

Mj.: Felhasználhatod a `.toUpperCase` és a `.slice` metódusokat, vagy az elemek stílusát módosítva is elérheted, hogy nagy kezdőbetűvel kezdődjenek!

Minta

Lista	Kepek	Diagram
-------	-------	---------

Bevásárló lista

- **Sajt** - **2849**
- Szalonna - 1637
- Tejföl - 469
- **Kávé** - **2199**
- Paprika - 708

1. Összes költség: 7862
2. Van-e 0-ra végződő ár? Nem
3. A legdrágább termék neve: Sajt

Képek



Diagram

Sávdiagram a termékek árairól!

