

Mozi

Kurityák Dániel

Készítse el az egy moziról szóló weboldalt a leírás és a minta alapján! A JavaScript kódját a `src/js/main.js` fájlba írja! Amennyiben lehetséges, használjon destrukciót és tömbfüggvényeket!

1. Másolja le a `forras` mappát `mozi-vezeteknev-keresztnev` néven, ahol a `vezeteknev-keresztnev` helyére a saját nevét írja! Az így létrehozott projektmappát nyissa meg.
2. Az oldal láblécébe helyezze el a saját nevét!
3. Hozzon létre egy `createSpan(text, class)` nevű függvényt, melynek feladata egy `span` elem létrehozása. A függvény első paramétere a megjeleníteni kívánt szöveg, a második az elemhez hozzáadandó osztályok tömbje.
 - a. A függvényben hozzon létre egy `span` elemet, majd állítsa be neki a paraméterként kapott szöveget!
 - b. Destrukció segítségével adja hozzá a paraméterül kapott osztályokat a tömbhöz! Gondoskodjon róla, hogy ha nincs megadva a paraméter, akkor ne keletkezzen hiba!
 - c. A függvény adja vissza az így készült `span` elemet!
4. Az előző feladatban létrehozott függvény segítségével hozza létre az alábbi `span` elemeket a mintán látható szöveggel...
 - a. A `film-count` azonosítójú elembe a `films` tömb elemszámát, az elem kapja meg a `text-lg` osztályt!
 - b. A `film-count` azonosítójú elembe a `films` tömb elemszámát szorozza meg 120-al, az elem kapja meg a `text-lg` és `astrict` osztályokat!
 - c. Destrukció segítségével bontsa elemére a legolcsóbb jegy adatait a `tickets` tömbből (tipp: rendezze a jegyeket növekvő sorrendbe). Az így kapott három adatot használja fel az alábbiak szerint:
 - A `cheapest-ticket-price` azonosítójú elembe a jegy árát, az elem kapja meg a `text-lg` és `double-astrict` osztályokat!
 - A `cheapest-ticket-details` azonosítójú elembe (a felsorolás a lap alján) *végén* jelenítse meg a jegy nevét és típusát!
5. A jegyek neve nem nagybetűvel kezdődik. Hozzon létre egy `capitalize(string)` nevű függvényt!
 - a. Destrukció segítségével mentse ki az első karakterét a `string`nek, illetve a maradék karakterek halmazát külön-külön változókba!
 - b. Alakítsa nagybetűssé az első karaktert, majd fűzze mögé a `string` többi karakterét is (tipp: a “maradék elemek” típusa tömb, így használja a megfelelő tömbfüggvényt a konvertálásukra)!
6. Készítse el a `createTicketCard(name, type, price)` függvényt, amely egy adott jegyhez hoz létre egy kártyát az alábbiak szerint:
 - a. Hozzon létre egy `div` elemet, amely kapja meg a `ticket-card` osztályt! Ez lesz a kártya törzse.
 - b. Hozzon létre egy `h3` elemet, amely kapja meg az előző feladatban létrehozott függvény alkalmazásával a jegy nevét!
 - c. Hozzon létre egy `p` típusú elemet, amely kapja meg a jegy típusát, valamint a `ticket-type` osztályt!
 - d. Hozzon létre egy `p` típusú elemet, amely kapja meg a jegy árát a *forint* szöveggel, valamint a `ticket-price` osztályt!
 - e. Adja hozzá a `h3` és két `p` elemeket a `div` elemhez, majd legyen az a függvény visszatérési értéke!
7. Egészítse ki a `generateTicketCards()` függvényt az alábbiak szerint:
 - a. Menjen végig a `tickets` elemein, majd hozzon létre minden jegyhez egy kártyát az előző feladatban létrehozott függvénnyel! A paraméter megadásához használjon destrukciót!
 - b. Az így kapott kártyát adja hozzá a `cards` tömbhöz!
 - c. Az összes kártya legenerálása után a `tickets` azonosítójú elem gyermekeit cserélje le a `cards` tömb tartalmára! Használjon destrukciót!

8. Készítse el a `createFilmCard(name, image)` függvényt, amely egy adott filmhez hoz létre egy kártyát az alábbiak szerint:
- Hozzon létre egy `div` elemet, amely kapja meg a `film-card` osztályt! Ez lesz a kártya törzse.
 - Hozzon létre egy `img` elemet, amelynek az `src` attribútuma legyen a paraméterül kapott kép, alternatív szövege a film neve!
 - Hozzon létre egy `h3` elemet, amely kapja meg a film nevét!
 - Adja hozzá az `img` és a `h3` elemeket a `div` elemhez, majd legyen az a függvény visszatérési értéke!

9. Egészítse ki a `generateFilmCards()` függvényt az alábbiak szerint:

A filmekhez tartozó képek elnevezése megegyezik a tömbben tárolt adatokkal, azzal a különbséggel, hogy szóközők helyett alsóvonalat (`_`) tartalmaznak (valamint `.jpeg` kiterjesztésűek, a `src/assets/images` mappában).

- a. Alakítsa át a `mappedFilms` változóba a `films` tömböt úgy, hogy a tömb egy eleme álljon egy másik tömbből, amelynek 0. indexén a film neve, 1. indexén a filmhez tartozó kép elérési útvonala van, például:

```
[  
  ["Oppenheimer", "src/assets/images/Oppenheimer.jpeg"],  
  ["Egy Minecraft film", "src/assets/images/Egy_Minecraft_film.jpeg"],  
]
```

- b. A mozi ki szeretné emelni a három legfelkapottabb filmjét. Vágja ki a `mappedFilms` tömbből az első három elemet, majd ezeket adja hozzá a `cards` tömbhöz! Használjon destrukciót **már a for ciklusban!**
- c. Az összes kártya legenerálása után a `films` azonosítójú elem gyermekeit cserélje le a `cards` tömb tartalmára! Használjon destrukciót!
- d. A maradék filmeknek csak a képüket kell megjeleníteni. Menjen végig a `mappedFilms` maradék filmjein és hozzon mindenegyhez létre egy `img` elemet, melynek állítsa be, hogy az `src` attribútuma a kép elérési útvonala legyen! Az így kapott elemet adja hozzá az `images` tömbhöz!
- e. Az összes kép legenerálása után az `other-films` azonosítójú elem gyermekeit cserélje le az `images` tömb tartalmára! Használjon destrukciót!