

# C csoport

## irodalom\_mufaj

### A feladatrész

0. Hozz létre egy index.html-t és illeszd be az alábbi tartalmat:

<https://gist.github.com/gomszab/127067bb3c776d45370db9d241d0757e> 0 pont

1. A táblázatból csinálj 2 táblázatot! Az egyiknél összevont soros cellák, a másiknál összevont oszlopos cellák legyenek.

Az összevont soros cellákkal rendelkező táblázat képzése a következő: Hagyd el az utolsó oszlopot, amelynek nincs a fejlében szöveg, és azoknál a soroknál, ahol megegyezik az első oszlop, állíts be sor összevonást, és törlök a felesleges cellákat!

Az összevont oszlopos cellák képzése a következő: Az üres cellákat tüntesd el oszlopok összevonásával.

Az összevont soros összevonást htmlben valósítsd meg, míg a cella oszlopos összevonását javascriptben egy adattömb vezetésével! 20 pont

2. Vezess be egy lenyíló menüt **tableselector** azonosítóval a html oldalon. A lenyíló menü 2 opcióval rendelkezik, amelyek segítségével az aktuális div elemet meg tudjuk jeleníteni. Az opciók értékei: **htmlsection**, illetve **jssection**. Alapértelmezetten a **jssection** elem van kiválasztva. A nem aktív táblázat elrejtéséhez helyezd bele a táblázatokat 1-1 divbe melyek azonosítója **htmlsection** illetve **jssection**, és használd az előre definiált **hide** css szabályt a létrehozott elemeken. 10 pont

3. Commitold a munkádat „tablazatok” üzenettel. **Ha nincs meg, automatikusan 0 pont a dolgozat.**

4. Vezess be űrlapot **htmlform** azonosítóval a html táblázathoz, amelynek a táblázat megadható értékei alapján vegyél fel beviteli mezőket és egy gombot, amivel el tudjuk küldeni az adatokat. A beviteli mezők azonosítói az oszlopszáma legyen magyarul ékezetek nélkül. Lehetséges azonosítók: **elso**, **masodik**, **harmadik**, **negyedik**, **otodik**. Az azonosítóknál balról jobbra haladunk, és a táblázat alapján opcionális értékek rendelkeznek magasabb azonosítóval. pl.: Műfaj oszlopa: **elso**, Mű2 oszlopa: **otodik**. A beviteli mezők név attribútuma utaljon az adatra, amit beírunk pl.: **mufaj**. Az űrlap implementálásakor minden beviteli mezőt a hozzá tartozó label-el (ami egyértelműen azonosíthatóvá teszi a felhasználó számára, hogy milyen adatot vár az alkalmazás) helyezz el egy div elemben és a beviteli mező alatt legyen egy html elem, ami a validáció esetén a hibaüzenetet jeleníti meg, és rendelj hozzájuk **error** css osztályt. Az eseménykezelés részét **TILOS EBBEN A FELADATRÉSBEN** implementálni. Az űrlapnak csak akkor kell megjelennie az oldalon, ha a hozzá tartozó táblázat is megjelenik. A debugolást megkönyíti, ha sortörésekkel teszel a beviteli mezők elő. 15 pont

5. Vezess be egy javascriptben implementált űrlapot a javascript táblázathoz **jsform** azonosítóval, amelynek a táblázat megadható értékei alapján vegyél fel beviteli mezőket és egy gombot, amivel el tudjuk küldeni az adatokat. A beviteli mezők azonosítói az oszlopszáma legyen magyarul ékezetek nélkül. Lehetséges azonosítók: **elso**, **masodik**, **harmadik**, **negyedik**, **otodik**. Az azonosítóknál balról jobbra haladunk, és a táblázat alapján opcionális értékek rendelkeznek magasabb azonosítóval. pl.: Műfaj oszlopa: **elso**, Mű2 oszlopa: **negyedik**. A beviteli mezők név attribútuma utaljon az adatra, amit beírunk pl.: **szerzo**. Az űrlap implementálásakor minden beviteli mezőt a hozzá tartozó label-el (ami egyértelműen azonosíthatóvá teszi a felhasználó számára, hogy milyen adatot vár az alkalmazás) helyezz el egy div semantikus elemben és a beviteli mező alatt legyen egy html elem, ami a validáció esetén a hibaüzenetet jeleníti meg, és rendelj hozzájuk **error** css osztályt. Az eseménykezelés részét **TILOS EBBEN A FELADATRÉSBEN** implementálni. Az

űrlapnak csak akkor kell megjelennie az oldalon, ha a hozzá tartozó táblázat is megjelenik. A debugolást megkönnyíti, ha sortöréseket teszel a beviteli mezők előtt. 15 pont

6. Commitolt az elkészült munkádat „form” üzenettel. **Ha nincs meg, automatikusan 0 pont a dolgozat.**

## B feladatrész

1. Implementáld az eseménykezelőt a **html-ben megvalósított űrlaphoz!** Vezess be azonosítót a html táblázat törzséhez **htmltbody** néven. Az eseménykezelő validálja a kötelező elemeket: Műfaj, Szerző, Mű. Ha a validáció nem sikeres, jelenítsünk meg hibaüzenetet a beviteli mező környékén, és ne füzzünk hozzá új sort a táblázathoz. Ha a validáció sikeres volt, akkor tüntessük el a hibaüzeneteket, törljük a beviteli mezők tartalmát, és füzzünk hozzá egy vagy, ha megadta az opcionális értékeket, két új sort a táblázathoz. 20 pont

2. Implementáld az eseménykezelőt a **javascript-ben megvalósított űrlaphoz**, amely hozzáfűz az adattömbhöz egy elemet, és a tömb alapján frissíti a táblázatot. Vezess be a javascript táblázat törzséhez azonosítót **jstbody** néven. Az eseménykezelő ellenőrizze a következő elemek tartalmát: Műfaj, Szerző, Mű. Ha a validáció nem sikeres, jelenítsünk meg hibaüzenetet a beviteli mező környékén, és ne adjunk hozzá új elemet az adattömbhöz. Ha a validáció sikeres volt, akkor tüntessük el a hibaüzeneteket, törljük a beviteli mezők tartalmát, füzzük hozzá az új elemet a adattömbhöz, és rendereljük újra a táblázat törzsét adattömb alapján. 20 pont

3. Commitolt az elkészült munkádat „befejezve” üzenettel.

+ Implementáld a feltételes validációt!