

# Időjárás – Eseménykezelés (gy)



Az utóbbi időben különösen sok csapadék esett Magyarország területén. Készíts egy időjárás statisztikákkal foglalkozó weboldalt az alábbi feladatleírás szerint! A munkád során kizárólag a **script.js** forráskódot kell módosítanod!

Az említett JavaScript fájl két konstans tartalmaz. Az elsőben települések adatai vannak tárolva. Minden település esetén ismerjük annak nevét, és a mért csapadékmennyiséget egy rögzített naptól (milliméterben). A másik konstans szavazások eredményeit tárolja.

1. Az adott napon hullot csapadékmennyiségeket osztályközökbe soroljuk. Minden csoportba két mennyiség tartozik, és 0-tól indítjuk a számozást. Összesen így 5 kategóriát készítünk a minta szerint.

Az oldal tetején található „Kategóriák megjelenítése” feliratú gombra **kattintva** jelenjenek meg a lehetséges kategóriák egy legördülő lista elemeiként!

A kattintást követően a gomb ne legyen a továbbiakban használható, és ezt a felhasználó számára is jelezzük azzal, hogy inaktív állapotra állítjuk!

Kategóriák megjelenítése

Kategória kiválasztása

0 - 1 ▾  
0 - 1  
2 - 3  
4 - 5  
6 - 7  
8 - 9

Név:

2. Amikor kiválasztunk egy kategóriát, szeretnénk megjeleníteni a hozzá tartozó településeket egy számozatlan felsorolásban.
  - a. Készíts egy `listaGeneralas(also, felső)` nevű függvényt, amely paraméterként megkapja a kategória alsó és felső határát, visszatérési értéke pedig egy számozatlan felsorolás, amely lista elemként tartalmazza a kategóriába tartozó települések neveit és csapadékmennyiségét a minta szerint!

Kategória kiválasztása

6 - 7 ▾

Cegléd: 6 mm
Debrecen: 7 mm
Kecskemét: 6 mm
Budapest: 7 mm
Pécs: 6 mm

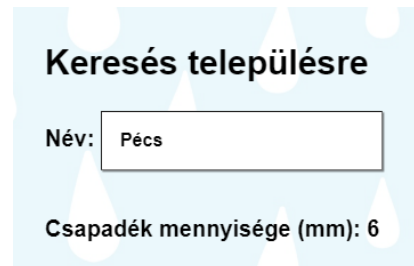
- b. A legördülő lista egy elemének kiválasztásakor (vagyis amikor értéke **változik**) olvasd ki a kategóriát, hozd létre a neki megfelelő listát, majd jelenítsd meg azt a **telepulesek** azonosítójú `<div>` elembe! Ügyelj rá, hogy egymás után többször is használható legyen a legördülő lista!

3. Szeretnénk keresni a weboldalon adott településre vonatkozóan csapadék adatokat.

- a. Készíts `keres(nev)` függvényt, amely paraméterként megkapja egy település nevét, visszatérési értéke pedig a települések lista hozzá tartozó település objektuma.

Amennyiben nem található ilyen nevű település, akkor a visszatérési érték tetszőleges lehet, de javasolt az `undefined` érték vagy `null` objektum használata.

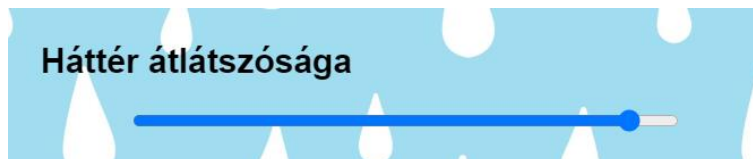
*(Feltehető, hogy a települések nevei egyediek, azonosítóként használhatók.)*



- b. A `kereses` azonosítójú szakaszban lévő beviteli mező tartalmát **módosítva** a hozzá tartozó bekezdésen belüli `<span>` elembe kerüljön bele a keresett település csapadékmennyisége! Ha a keresett település nem található, akkor egyetlen `”-,”` karakter legyen látható a megadott helyen!

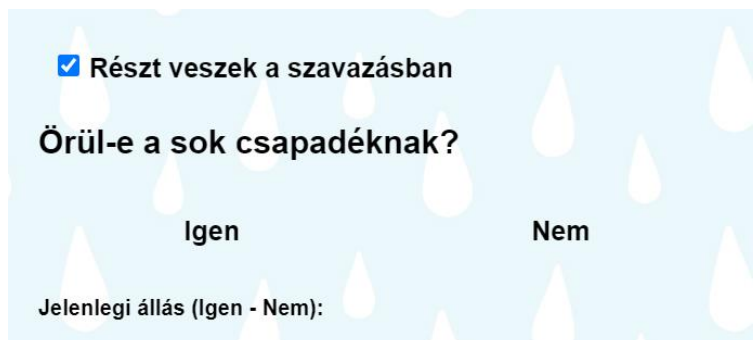
4. Az oldal háttere a `<body>` első gyermekeként elhelyezkedő `hatterkep` azonosítójú `<div>` elem által került megvalósításra. Ennek alapértelmezett átlátszósága 20%, amely a `style.css` fájlban beállításra került.

Az oldalon található egy csúszka, amely segítségével a látogató szabadon módosíthatja ezt az átlátszóság értéket. Ennek kezdőértéke szintén 0.2 az `index.html` állomány szerint.



Oldd meg, hogy a csúszka segítségével módosítani lehessen az oldal háttérének átlátszóságát! Minden **változáskor** az átlátszóság a csúszka aktuális értéke legyen!

5. Egy szavazás keretein belül a látogatók elmondhatják véleményüket az időjárásról. Oldd meg, hogy a `velemeney` azonosítójú szakasz pontosan akkor legyen látható, ha a felhasználó bejelöli a fölötte található checkbox elemet!



6. A szavazás úgy történik, hogy választani kell egy „Igen” és egy „Nem” feliratú radio gomb közül. Amelyikre **kattint** a felhasználó, arra szavazott. Ezután megjelenik a bekezdés végén a szavazás jelenlegi állása, az újként leadott szavazattal együtt!

- a. Készíts **arany(típus)** néven függvényt, amely paraméterként megkap egy szavazat típust („igen” vagy „nem”), visszatérési értéke pedig, hogy az összes szavazat hány százaléka tartozik a választott típusba. Ezt a százalék értéket egy tizedesre kerekítve adja meg!
- b. Valósíts meg egy olyan eseménykezelő függvényt, amely bármelyik radio gomb kiválasztása esetén helyesen működik, vagyis rögzíti a választás eredményét, majd megjeleníti a szavazás aktuális állását!



- c. Ügyelj rá, hogy csak egyszer lehessen szavazni, tehát ha a felhasználó rákattintott valamelyik radio gombra, akkor a továbbiakban azok már ne reagáljanak a kattintás eseményre! Ezt a felhasználó számára is jelezt a gomb letiltásával!
7. Sok felhasználó jelezte, hogy a települések keresésekor örülne valamilyen segítségnek, például ha megjelenítené a weboldal azok neveit.
- a. Amint **fókuszba kerül** a keresőmező, jelenjenek meg a kereshető települések nevei az alatta található számozatlan felsorolásban! A lista elemek létrejöttét követően az `<ul>` elem szegélye 1px vastag, szaggatott, szürke legyen!

- b. A keresőmezőt **elhagyva** (vagyis amikor már nincs fókuszban) tűnjön el a lista tartalma és szegélye egyaránt!

**Kitekintés: A többi feladat megoldásához szükséges tudás még nincs a birtokunkban!**

- c. Ugyanaz az eseménykezelő tartozzon az a) és b) feladatok eseményeihez!
- d. A segítségként felugró lista elemeire kattintva a megfelelő szöveges érték kerüljön be a keresőmezőbe! (Javasolt esemény: egér lenyomása.)
- e. Ügyeljünk rá, hogy a d) feladat kattintásakor a csapadék mennyiségét tartalmazó `<span>` elem tartalma is frissüljön, vagyis jelezzük, hogy megváltozott a keresőmező tartalma (idézzük elő az esemény bekövetkezését).

## Minta

Kategóriák megjelenítése

### Kategória kiválasztása

4 - 5 ▾

Albertirsa: 5 mm
Veszprém: 4 mm
Szolnok: 5 mm
Abony: 4 mm

---

### Keresés településre

Név:

Csapadék mennyisége (mm): -

---

### Háttér átlátszósága

---

☒ Részt veszek a szavazásban

### Örül-e a sok csapadéknak?

Igen

Jelenlegi állás (Igen - Nem):

15.1% - 84.9%