

# Budapest

(120 perc)

Egy igen neves, kimutatásokkal foglalkozó cég szeretne egy webes alkalmazást, ahol a Budapesti lakosok összetételét és annak változását lehetne nyomonkövetni. Ezen webalkalmazás elkészítésével bízzák meg önt.

A szükséges adatok rendelkezésre állnak egy külső backend szerveren. A szerver elérése a következő:

***<https://budapest-lakossaga-default-rtdb.europe-west1.firebaseio.com/>***

A szerveren a következő REST API függvények érhetőek el:

1.) /lakosok.json

Metódus: GET

Válasz: Az adatbázisban szereplő rekordok

Minta:

```
{
  "0": {
    "evszam": 1880,
    "magyar": 227894,
    "nemet": 138423,
    "szlovak": 25637,
    "egyeb": 10752
  },
  ...
  "legujabb": {
    "egyeb": 200191,
    "evszam": 2022,
    "magyar": 1908711,
    "nemet": 3319,
    "szlovak": 1919
  }
}
```

2.) /lakosok.json

Metódus: POST

A kérés törzsében kell a szerver felé elküldeni a felvenni kívánt új rekordot.

Minta:

```
{ "evszam": 2022,
  "magyar": 1908711,
  "nemet": 3319,
  "szlovak": 1919,
  "egyeb": 200191
}
```

## Feladatok:

- 1.) Készítsen új projektet az Ön által tanult keretrendszerben (Angular, React, VueJs, stb.) *BudapestPeldaBela* néven, ahol a *PeldaBela* helyett az Ön neve szerepel. Adja hozzá a projekthez a legfrissebb Bootstrap keretrendszert.
- 2.) Készítsen egy új komponenst, amely a webalkalmazás nyitóoldala lesz. A nyitóoldal feleljen meg a következő elvárásoknak:
  - a. A nyitóoldal tartalmazza az *index.txt* fájlban található tartalmat maradék nélkül. Ügyeljen arra, hogy az egyes részek megfelelő HTML címkékkel legyenek ellátva (címsor, bekezdés, felsorolás, táblázat, stb.)
  - b. A nyitóoldalon foglaljon helyet az *budapest.png* kép.
  - c. A nyitóoldalon helyezzen el kettő navigációs menüpontot:
    - i. Lakosság a népszámlások alkalmával
    - ii. Új felmérés felvétele
- 3.) Hozzon létre új komponenst, amelyen a népszámlálási adatok megjelenhetnek.
  - a. Az adatok a kiszolgáló szerverről a megfelelő REST API függvénnyel kérhető le.
  - b. Az adatokat jelentse meg táblázatos formában, évszám szerinti csökkenő sorrendben. Ügyeljen arra, hogy az oszlopok sorrendje a következő legyen: évszám, magyar, német, szlovák, egyéb lakosság. (A táblázatkészítéséhez használhat külső komponenst is, pl.: Angular Material, PrimeNG, Quasar, stb.)
  - c. Állítsa be a komponensnek az eléréséhez a */adatok* útvonalat.
  - d. A kezdőoldalon a megfelelő feliratú linknek/gombnak úgy állítsa be a navigációját hogy az a */adatok* útvonalra navigáljon.
- 4.) Hozzon létre egy újabb komponenst, amely segítségével új felmérést lehet regisztrálni.
  - a. A komponens elkészítéséhez használjon űrlapot, benne megfelelő beviteli mezőkkel. Ügyeljen arra, hogy a beviteli mezőkbe csak valós értékeket lehessen beírni.
  - b. Az évszám egy szerkeszthető mező. Viszont az oldal betöltődésekor mindig az aktuális évszám szerepeljen benne.
  - c. Az űrlap elküldésével az adatok kerüljenek fel a szerverre a fentebbi API leírás alapján.
  - d. Sikeres küldés esetén navigáljon a */adatok* útvonalra.
  - e. Ezt a komponenst az */ujadat* útvonalon lehessen elérni. A kezdőoldalon a megfelelő gomb erre az útvonalra irányítson.

- 5.) Készítsen hibakezelést. A REST API metódushívások esetlegesen felmerülő hibáinak elkapására, megjelenítésére. A hibakódot és üzenetet jól látható részen és jól látható módon jelenítse meg.
- 6.) A weboldal megjelenésének kialakítására használhat Bootstrap osztályokat, vagy saját maga által készített (és tanult) CSS osztályokat is. Ügyeljen a megjelenés kialakításánál a reszponzivitásra!
- 7.) Elkészült projektjének forráskódját a választott technológiának megfelelően adja le:
  - a. Törölje a felesleges, nem használt állományokat
  - b. A forráskódot tömörítse .zip formátumba, Vezetéknév\_Keresztnév.zip néven.
  - c. A tömörített fájlt adja be/töltse fel a kívánt helyre.