

Készíts egy projektet egy egyszerű webshophoz. A projekt legyen képes megjeleníteni a termékeket a főoldalon. Legyen lehetőség regisztrálni, illetve a regisztrált adatokkal bejelentkezni. Bejelentkezés után legyen lehetőség új terméket is felvenni, illetve kijelentkezni. Minden felvett adatot egy adatbázisban tárolj el, kivétel a termékek képét, amit egy külön mappába helyezz el. Minden formba bevitt adat helyességét kliens és szerver oldalon is ellenőrizd. A feladat megoldásához használd segítségül a korábbi projekteket és az alábbi feladatléírást. A beadás ideje és módjával kapcsolatban kövesd a gyakorlatvezető utasításait.

A feladatban kép tárolása és megjelenítése is szerepel, viszont mivel ezt nem tanultuk így ez szorgalmi feladat. Amennyiben nem kerül kép eltárolásra úgy a kártyák fejlécében a termék neve szerepeljen, és a termékek listázását JavaScripttel végezd.

Beadás módja: github repo link. A github repo neve SajjatNev\_webshop formájú legyen. pl.: Ha a neved Gipsz Jakab Sándor, akkor a repo neve GipszJakabSandor\_webshop legyen. A github repoban mellékelj az adatbázis exportját is.

## 1.Feladat Készíts egy adatbázist **webshop** néven.

a.) Hozz létre egy **felhasznalo** táblát az alábbi mezőkkel, a mezők egyike sem vehet fel nullt értékül:

- id – Egész, elsődleges kulcs, autoincrement
- felhasznalo\_nev – Szöveges, max 30 karakter, egyedi
- email – Szöveges, max 255 karakter, egyedi\*
  - Az adatbázis tárolómotortól és verziótól függően eltérő maximális mérete lehet az egyedi indexeknek. InnoDB esetén régebbi verzióknál ez 767 Byte. Kompatibilitás miatt az adatbázis karakterkészletét érdemes úgy kiválasztani, hogy 1 karakter maximum 3 byteot foglaljon. (javasolt utf8\_hungarian\_ci illesztés használata az utf8mb4-es verzió helyett)
- jelszo – Szöveges, max 100 karakter
- teljes\_nev – Szöveges, max 100 karakter
- szuletesi\_datum – Dátum
- irányito\_szam – Egész, 4 számjegy hosszú
- varos – Szöveges, max 50 karakter
- cím – Szöveges, max 255 karakter
- regisztracio\_idopontja – Dátumidő, alapértelmezetten a rekord létrehozásának időpontja.

b.) Hozz létre egy **termek** táblát az alábbi mezőkkel:

- id – Egész, elsődleges kulcs, autoincrement
- nev – Szöveges, max 100 karakter, nem üres
- leiras – Szöveges
- ar – Egész, nem üres
- kep – Szöveges

- A képet fájlként tároljuk majd a szerveren, ennek a fájlnek legyen az elérési útvonala
  - c.) Az adatbázist létrehozó scriptet webshop.sql néven mentsd el a projekt rootjában.
- 2.Feladat** Készíts egy webalkalmazást a webshop számára.
- a.) A projekt 4 oldalból álljon: **Főoldal, regisztráció, bejelentkezés és termékek hozzáadása.**
  - b.) Az oldalakat bootstrapes osztályok segítségével formázd.
  - c.) Minden oldal az oldal tetején tartalmazzon egy navigációt, amely mind a 4 oldalhoz tartalmaz hivatkozást, illetve egy kijelentkezés gombot is.
    - A regisztráció és bejelentkezés hivatkozások csak kijelentkezett állapotban legyenek láthatók.
    - A termékek hozzáadása hivatkozás és a kijelentkezés gomb csak bejelentkezett állapotban legyen elérhető
    - Az éppen aktuális oldal hivatkozása kapja meg az active osztályt, ezáltal a kinézete térjen el a többi hivatkozástól.
  - + A regisztráció, bejelentkezés és kijelentkezés gombok a navigáció jobb oldalán helyezkedjenek el.
  - d.) A főoldal:
    - Jelenítse meg az összes adatbázisban szereplő terméket
    - Az elemek kártyákban helyezkedjenek el egy sorban úgy, hogy egy sorba kijelző méretétől függően 4, 3, 2 vagy 1 termék helyezkedjen el.
    - Bejelentkezett állapotban a kinézet a webshop\_minta.png-hez hasonló legyen.
  - e.) Regisztráció
    - Az oldal tartalmazzon egy bootstrapes formot, amelyet úgy készíts el, hogy a felhasználó tábla minden oszlopa értéket kaphasson.
    - A beviteli mezők típusát úgy add meg, hogy azzal korlátozni lehessen, hogy a felhasználó megfelelő adatot adjon meg.
    - A beviteli mezők mindegyike legyen kötelező.
    - Ahol szükséges, ott az input mezőkbe bevitt adatok hosszát korlátozd megfelelő attribútummal.
    - Ügyelj rá, hogy a jelszó legalább 8 karakter hosszú legyen, tartalmazzon kis- és nagybetűt, valamint számot.
    - Az űrlap elküldése után szerveren oldalon is ellenőrizd, hogy a bevitt adatok megfelelnek-e a feltételeknek.
    - Ha minden adat megfelelően lett kitöltve, akkor ellenőrizd, hogy az adott felhasználónév / email cím ne szerepeljen az adatbázisban.
    - Ha nincs duplikált adat, akkor add hozzá az adatbázishoz az új felhasználót.
      - Ügyelj rá, hogy a jelszót ne olvasható formában, hanem annak egy biztonságos hashelési algoritmussal (bcrypt, argon2) létrehozott hashét tárold el az adatbázisban.
    - A regisztráció után annak sikerességéről a regisztrációs űrlap felett bootstrapes alertben adj visszajelzést.

- Siker esetén az a figyelmeztetés alert-success osztályú legyen, sikertelen regisztráció esetén pedig alert-danger osztályú.
  - Sikertelen regisztráció esetén töltsd újra az űrlapot a felhasználó által beírt adatokkal
  - Sikertelen regisztráció esetén írd ki, hogy mi okozta annak sikertelenségét.
- + Amennyiben az e-mail vagy felhasználónév már foglalt, akkor még az űrlap elküldése előtt figyelmeztesse az oldal a felhasználót és az űrlapot ne lehessen elküldeni.
- f.) Bejelentkezés
- Az oldal tartalmazzon egy bootstrapes formot, két beviteli mezővel: felhasználónév és jelszó.
  - A beviteli mezők kitöltése legyen kötelező, viszont további ellenőrzéseket kliens oldalon ne hajts végre.
  - Az űrlap elküldése után ellenőrizd az adatok helyességét.
  - Helyes adatok esetén vizsgáld meg, hogy az adatbázisban valóban szerepel-e az adott felhasználónév és jelszó kombinációval felhasználó
    - php-ban a beírt jelszó és az eltárolt hash összehasonlításához a password\_verify függvényt alkalmazható.
  - Amennyiben jó kombinációt adott meg a felhasználó, úgy a felhasználó azonosítót tárold el a munkamenetben, majd irányítsd át a felhasználót a főoldalra
  - Sikertelen bejelentkezés esetén alert-danger osztályú bootstrapes alertben figyelmeztess erről a felhasználót a bejelentkezés űrlap felett. A hiba okát itt ne jelöld meg.
- + A bejelentkezési oldalon ne csak felhasználónevet, hanem felhasználónevet vagy e-mail címet lehessen megadni.
- g.) Termék hozzáadása
- Az oldal tartalmazzon egy bootstrapes formot, amelyet úgy készíts el, hogy a termék tábla minden oszlopa értéket kaphasson.
  - A beviteli mezők típusát úgy add meg, hogy azzal korlátozni lehessen, hogy a felhasználó megfelelő adatot adjon meg.
  - A leírás kivételével minden beviteli mező legyen kötelező.
  - Ahol szükséges, ott az input mezőkbe bevitt adatok hosszát korlátozd megfelelő attribútummal.
  - A leírás számára input helyett textarea-t használj.
  - Az ár input mezőnél ne lehessen negatív számot értékül adni.
  - A kép beviteli mezőnek mindenképp egy képfájl kelljen megadni. Ezt a file input accept tulajdonságával lehet megadni.
  - Ügyelj rá, hogy az űrlap alkalmas legyen fájl feltöltéséhez.
  - Az űrlap elküldése után az adatokat szerver oldalon is validáld:
    - Csak akkor haladj tovább, ha a név nem üres, és az árnak nemnegatív számot adott meg.

- Ellenőrizd, hogy a kép számára valóban kép került felköltsre. (MIME típus image szóval kezdődik)
  - A feltöltött képet egy létrehozott uploads könyvtárba rakd bele. A kép neve a termék nevét és a feltöltés időpontját tartalmazza.
    - Az adatbázisba az így feltöltött kép neve kerüljön be.
  - A hozzáadás után annak sikerességéről a regisztrációs űrlap felett bootstrapes alertben adj visszajelzést.
    - Siker esetén az a figyelmeztetés alert-success osztályú legyen, sikertelen felvétel esetén pedig alert-danger osztályú.
    - Sikertelen felvétel esetén írd ki, hogy mi okozta annak sikertelenségét.
- h.) Kijelentkezés
- A gombra kattintva töröld az aktuális munkamenetet majd irányítsd át a felhasználót a főoldalra.