


## Jegymegajánló ZH 1.

Készíts MySQL adatbázist a következő szerkezettel (adatbázis neve: „adatok” tábla neve: „tabla”):

Sor 	Username	Titkos
1	katika@gmail.com	piros
2	arpi40@freemail.hu	zold
3	zsanettka@hotmail.com	sarga
4	hatizsak@protonmail.com	kek
5	terpeszterez@citromail.hu	fekete
6	nagysanyi@gmail.hu	feher

*(nem használtam ékezeteket, hogy véletlenül se legyen kódolási probléma)*

Készíts egy **esztétikus** PHP oldalt, amely egy formban bekéri a felhasználói nevet (Username) és a jelszót (echo nélkül). A bevitt információkat GET vagy POST tömb segítségével juttasd át a szerver oldalra.

A szerver oldalon található egy password.txt állomány (mellékelve). A fileban az adatbázisban is megtalálható Username található egy \* karakter majd a hozzá tartozó jelszó. A sorok szabványos Linux EOL -al vannak lezárva azaz csak egy 0A byte-al (azaz 13-as karakter nincs) pl.:

```
katika@gmail.com*jelszó1
arpi40@freemail.hu*jelszó2
zsanettka@hotmail.com*jelszó3
...
```

Természetesen a valódi, titkos jelszavak vannak az állományban.

A password.txt-ben azért nem látjuk a fenti sorokat közvetlenül, mert le vannak titkosítva. Egy jelszó állományt illik ugyanis titkosítani. A titkosítási algoritmus nagyon egyszerű, mégis hatékony. A karakterek bájtonként egy-egy számértékkel vannak eltolva. A megoldó kulcs : 5,-14,31,-9,3 amely értékekkel el lettek tolvá a karakterek kódjai. Ezekkel a számokkal kell „visszatekerni” a karaktereket sorba az 1. - 5. számig, majd ismét az 1. -től kezdve körbe – körbe. Az EOL természetesen nincs kódolva.

A PHP program tehát azonosítsa, hogy a form-ba beolvasott felhasználói névhez a (password.txt alapján) megfelelő jelszót adták-e meg. Ha igen akkor az adatbázisból a megfelelő színű korong vagy egyéb kép állomány jelenjen meg. Amennyiben rossz felhasználói nevet adott meg a „nincs ilyen felhasználó” hibaüzenetet írja ki. Ha rossz jelszót adott meg akkor „hibás jelszó” üzenetet adjon és 3 másodperc múlva dobja át a felhasználót a police.hu -ra.

A helyes jelszavakat felesleges megadnom, hiszen benne vannak a password.txt-ben és ha a felhasználó nevet ki tudta kódolni, akkor a jelszavakat is. Így tudod tesztelni is a programod helyességét.

Nem nagy kódú, egyszerű, gyorsan megírható mégis sok kompetenciát és hasznos technológiákat igénylő feladat. Ha nincs teljesen kész akkor is értékelhető és alkalmas a megfelelő differenciálásra (1-5 jegy).

Fontos még: A program írja ki, véleményed szerint milyen jegyet érdemelsz! Komolyan! De csak realisan!

Internet használható, más személy vagy előre megírt program nem!

Az értékelést a három csoport összes megoldásai alapján fogom differenciálni (azaz egymáshoz fogom viszonyítani a megoldásokat és az alapján alakul ki a színvonal). Így ha túl nehéz lenne a feladat és sok a nem szép, hiányos vagy rossz megoldás, akkor értelemszerűen engedékenyebb az értékelés is (feladatnehézség alapján, önmagát optimalizáló értékelési rendszer).

Ebből az is következik, hogy ha pl. a 2. , 3. csoport már ismeri a feladatot és emiatt jobb megoldásokat ad be, akkor azzal feljebb viszi az értékelési szintet és a feladatot kiszivárogtató korábbi csoportok eredményei romlanak ezáltal. Tehát magadnak csinálod a bajt!

Ne legyen két egyforma megoldás, azonnal észrevehető ha két program ugyan olyan...

A feladatra adott idő: 3 óra

Hamarabb is be lehet fejezni.

A megoldásokat (lehetőleg egyetlen php állomány, a színes képek linkkel hivatkozva az internetre – nem csatolva) a következő e-mail címre küldjétek el: [hunortaltos@gmail.com](mailto:hunortaltos@gmail.com) Tárgy: Név – Neptunkód

Az adatbázist nem kell elküldeni, de a szerkezete pontosan a fenti legyen (DB név, táblanév, mezők pontosan egyezzenek meg karakterről-karakterre) a programban hivatkozva rá, a kipróbálhatóság miatt. Tehát a php-t letöltve a gépemre a megfelelő DB mellett rögtön működjön.