

Szkriptnyelvek - JavaScript ismertető

- A programot JavaScript nyelven kell megírni.
- A benyújtandó fájl neve: `feladat.js`
 - Egy JavaScript nyelven írt, szöveges fájl (nem zip, rar, stb.)
 - Ez csak a feladatban kért dolgokat tartalmazza! Amennyiben saját inputtal teszteled a kódot lokálisan, úgy feltöltés előtt a tesztelő kódrészletet kommenteld ki!
- A megoldást Bíró2 webes felületén (<https://biro2.inf.u-szeged.hu>) keresztül kell benyújtani és a megoldást a Bíró fogja kiértékelni.
 - A Feladat beadása felületen a *Feltöltés* gomb megnyomása után ki kell várni, amíg lefut a kiértékelés. **Kiértékelés közben nem szabad az oldalt frissíteni vagy a Feltöltés gombot újból megnyomni** különben feltöltési lehetőség veszik el!
- Feltöltés után a Bíró a programot **Node** interpreterrel fogja futtatni, és különböző tesztesetekre futtatja.
- A program működése akkor helyes, ha a tesztesetek futása nem tart tovább 2 másodpercnél és hiba nélkül fejeződik be, valamint a program működése a feladatkiírásnak megfelelő.
- Ha 3 teszteset futási ideje túllépi a fenti időkorlátot, a tesztelés befejeződik, a pontszám az addig szerzett pontszám lesz.
- A riport.txt megtekinthető az alábbi módon:
 1. Az Eredmények megtekintése felületen a vizsgálandó próba új lapon való megnyitása
 2. A kapott url formátuma:
https://biro2.inf.u-szeged.hu/Hallg/IB370G/FELADAT_SZAMA/hXXXXXX/4/riport.txt
 3. Az url-ből visszatörölve a 4-esig (riport.txt törlése) megkaphatók a 4-es próbálkozás adatai
- A programot 25 alkalommal lehet benyújtani, a megadott határidőig.
- A munkád során figyelj arra, hogy pontosan kövesd a feladatban leírtakat, az elnevezéseket!
- A fájl elejére kommentbe írd be a neved, Neptun és h-s azonosítód az alábbi formában:

```
// Nev: vezeteknev keresztnév  
// Neptun: NEP4LF  
// h: h123456
```

Szkriptnyelvek - 2. ZH

1. feladat: Hungi (15 pont)

Írj JavaScriptben egy `hungi` nevű függvényt, amely tömböt vár paraméterben! Amennyiben undefined a paraméter térjünk vissza 0-val.

Ha a kapott tömb hossza kisebb mint 10, a függvény az alábbi szöveggel térjen vissza: `Tu1 kevesen jöttek a buliba!`

Ha a kapott tömb megfelelő hosszú, de a hossz `páratlan`, akkor a függvény az alábbi szöveggel térjen vissza: `A bulizok között biztos van egy szingli!`

Egyéb esetben a térjünk vissza egy olyan tömbbel, ami a tömbben található költségek közül az

átlag feletti költséket tartalmazza.
Mindig a tömb elejéhez fűzzünk hozzá!

Példa:

Input: [33333,300,2589,990,3780]
Return: Tul kevesen jettek a buliba

Input: [3333,300,258,9,990,3780,3333,300,2589,990,3780]
Return: A bulizok kozott biztos van egy szingli

Input: [3333,300,2589,990,3780,3333,300,2589,990,3780]
Return: [3780, 2589, 3333, 3780, 2589, 3333]

Input: undefined
Return: 0

2. feladat: Rémszarvas (30 pont)

- Írj JavaScriptben egy `Rémszarvas` nevű osztályt, amely a Krampusz egyik rémszarvasát reprezentálja! Az osztálynak legyen 3 adattagja: `_nev`, `tulajdonsagok` (a tulajdonságokat tároló tömb) és `remisztoseg`.
- A konstruktor 2 paramétert vár, a rémszarvas nevét és rémisztőségét. Inicializáld a `_nev` és a `remisztoseg` adattagot a konstruktor paraméterével! A `tulajdonsagok` adattag üres tömbként legyen inicializálva!
 - A konstruktorban az utolsó paraméter értékét (`remisztoseg`) ne legyen kötelező megadni, alapértéke legyen a 33!
- Készíts get és set property-t a `_nev` adattaghoz, `nev` néven! A getter adja vissza az adattag értékét, a setter pedig ellenőrizze, hogy a paraméterben kapott érték egy **legalább 4 karakter hosszú string** legyen!
 - Ha a setterben a paraméter típusa és értéke megfelelő, akkor állítsd be az adattagot a paraméterben kapott értékre!
 - Hibás típus vagy érték esetén a setter állítsa be az `Krampi` szövegre az adattag értékét!
- Írd át az osztály konstruktorát úgy, hogy a Rémszarvas neve itt is ugyanúgy legyen beállítva, mint a setterben!
- Írj egy `fejleszt` metódust, amely 1 paramétert vár, a tulajdonság nevét!
 - Ellenőrizd:
 - Az új tulajdonság típusa `string` -e? Amennyiben, nem akkor a `tulajdonsagok` tömb legyen üres.
 - Ha a tulajdonság string, akkor, ha egy szóból áll (nincs benne szóköz) szúrd be a `tulajdonsagok` tömb végére, ellenkező esetben szúrd be a `tulajdonsagok` tömb elejére.
- Írj egy paraméter nélküli `rosszindulat` metódust, amely visszatér a `tulajdonsagok` tömbben található tulajdonságok betűinek számainak átlagával! Ha a tömb üres, akkor 0-val térj vissza!

- Írj egy paraméter nélküli `info` metódust, amely a következő formátumú szöveggel tér vissza: `{nev} rémszarvasunk {remisztoseg} rémisztóséggel rendelkezik. Tulajdonságai: {tul}`, ahol a `tul` a `tulajdonsagok` tömbben található tulajdonságok szóközzel elválasztva!
- Írj egy `remiszt` paraméter nélküli metódust!
 - A függvény megnöveli 5-el a `remisztoseg`-et
 - A függvény igaz értékkel tér vissza, ha a `remisztoseg` nagyobb lett mint 50.
 - Ellenkező esetben hamis értékkel.

Jó munkát!