

Szkriptnyelvek - JavaScript minta ZH

1. feladat: Call of the wild (20 pont)

Írj egy `call` nevű függvényt, amely 2 paramétert vár: rendre egy callback függvényt és egy több szóból álló szöveget! A függvény a következőt csinálja:

- Ha valamelyik paraméter típusa nem megfelelő, akkor `null` értékkel tér vissza.
- Ha a második paraméterben kapott szöveg 2-nél kevesebb szóból áll, akkor `undefined` értékkel tér vissza.
- Minden egyéb esetben a függvény feldarabolja a második paraméterben kapott szöveget szóközök mentén, és a feldarabolás eredményeként kapott tömbbel meghívja az első paraméterben kapott callback függvényt. A függvény visszatérési értéke a callback függvény visszatérési értéke legyen!

Példa használat:

```
function wild(words) { return 'Szavak szama: ' + words.length; }
function lastWord(words) { return words[words.length - 1]; }

console.log(call(wild, 'JavaScript minta ZH'));           // 'Szavak szama: 3'
console.log(call(lastWord, 'JavaScript minta ZH'));      // 'ZH'
console.log(call('Hibas tipus', 'JavaScript minta ZH')); // null
console.log(call(wild, 'JavaScript'));                   // undefined
```

2. feladat: Beast of Gévaudan (30 pont)

Andris a közelgő ZH-kra való készülés helyett inkább épített egy időgépet, amivel visszautazott az 1700-as évekbeli Franciaország Gévaudan nevű tartományába. Sajnos Andris egy elég rossz időszakot választott az időutazásra, hiszen ebben az időben Gévaudan lakosságát egy ismeretlen eredetű `fenevad` tartotta rettegésben. A tartomány lakosai a fejlett technológiai ismeretekkel rendelkező Andrist kérték meg a bestia elkapására.

A feladat végrehajtásához Andris egy négyzetrácsos térképet rajzol a tartományról, amin van valamennyi sor, és a sorokon belül van valamennyi oszlopcella. A térképen azokat a cellákat jelöli be Andris, ahol már látták a fenevadat.

Írj egy `beastOfGevaudan` nevű függvényt, amelynek 3 paramétere sorban:

- `n`: a térképen szereplő sorok száma (egész szám)
- `m`: a térkép egyes sorain belül szereplő cellák száma (egész szám)
- egy szöveg, amely szóközzel elválasztott `x;y` koordináta-párokat tartalmaz - ezek a koordináták mondják meg, hogy a térkép mely területein látták korábban a fenevadat (`x` a sor, `y` pedig az oszlop indexét adja meg)!

Ha valamelyik paraméter típusa nem megfelelő, akkor a függvény egy üres tömbbel térjen vissza!

Megfelelő típusú paraméterek esetén a függvény térjen vissza egy `n` sorból és `m` oszlopból álló kétdimenziós tömbbel, amelyben a kapott koordinátáknak megfelelő indexeken `X` (nagy íksz) karakter szerepel, minden egyéb indexen pedig `-` (kötőjel) karakter áll! A függvény akkor is működjön hibamentesen, ha valamely koordinátával egy nem létező tömbindexre hivatkozunk (lásd: második példa)!

Példa a függvény működésére:

```
Input: 2, 3, '1;0 0;2 1;2'
```

```
Return: [['-', '-', 'X'], ['X', '-', 'X']]
```

```
Input: 3, 5, '2;0 0;3 2;4 10;1 2;3 0;0 2;1'
```

```
Return: [['X', '-', '-', 'X', '-'], ['- ', '- ', '- ', '- ', '- '], ['X', 'X', '-', 'X', 'X']]
```

```
Input: 'Hibas típus', 3, '1;0 0;2 1;2'
```

```
Return: []
```

3. feladat: Album (30 pont)

- Írj egy `Album` nevű osztályt, amely egy zenealbumot reprezentál! Az osztálynak legyen 3 adattagja: `_cim` (az album címe), `hossz` (az albumon található zenék összesített hossza másodpercben) és egy `zenek` nevű tömb, amely az albumon található zenék címét tárolja! **(1 pont)**
- A konstruktor egyetlen paramétert vár: az album címét, ami alapján inicializálja a `_cim` adattagot. A `hossz` adattag alpból 0-ra legyen inicializálva, míg a `zenek` adattagot egy üres tömbbel inicializálja a konstruktor!
 - A konstruktor paraméter értékét ne legyen kötelező megadni, alapértéke legyen az

Untitled szöveg! (4 pont)

- Készíts get és set property-t a `_cim` adattaghoz, `cim` néven! A getter adja vissza az adattag értékét, a setter pedig ellenőrizze, hogy a paraméter értéke egy **legalább 3 karakter hosszú szöveg** legyen!
 - Ha a setterben a paraméter típusa és értéke megfelelő, akkor állítsd be az adattagot a paraméterre!
 - Hibás típus vagy érték esetén a setter állítsa be a `Unknown` szövegre az adattag értékét! (5 pont)
- Írd át az osztály konstruktorát úgy, hogy a cím itt is ugyanúgy legyen beállítva, mint a setterben! (2 pont)
- Írj egy `zenetFelvesz` metódust, amely egy zenét reprezentáló objectet vár paraméterül! Az objectben a `zeneCim` property-hez a zene címe, míg a `zeneHossz` property-hez a zene (másodpercben megadott) hossza tartozik. A metódus szűrja be a zene címét a `zenek` tömb végére, és növelje meg a `hossz` adattag értékét a zene hosszával!
 - Amennyiben a paraméterben nem egy object érkezik, vagy az objectből hiányzik a `zeneCim` és `zeneHossz` property-k valamelyike, akkor a metódus ne csináljon semmit! (6 pont)
- Írj egy paraméter nélküli `albumHossz` metódust, amely átváltja a `hossz` adattagban lévő, másodpercben megadott hosszt percekre és másodpercekre! A metódus adja vissza az átváltott hosszt egy `{percekSzama} perc, {masodpercekSzama} masodperc` formátumú szövegben! (3 pont)
- Írj egy paraméter nélküli `info` metódust, amely a következő formátumú szöveggel térjen vissza: `{_cim} album, {zenekSzama} darab zenevel` (ahol `zenekSzama` a `zenek` tömb hossza)! (2 pont)
- Írj egy `osszehasonlit` metódust, amely egy `Album` objektumot vár paraméterül! A metódus adja vissza, hogy a jelenlegi albumnak és a paraméterben kapott albumnak az összes adattagja rendre megegyezik-e!
 - Ha a paraméterben nem egy `Album` típusú objektum érkezik, akkor a metódus `false` -szal térjen vissza!
 - Segítség a tömbök összehasonlításához: [link](#). (5 pont)
- Írj egy paraméter nélküli `randomZene` metódust, amely visszaadja a `zenek` tömb egy véletlenszerű elemét! A véletlenszám-generálás során ügyelj arra, hogy a tömb legelső és legutolsó elemét is vissza kell tudni adni!
 - Ha a `zenek` tömb üres, akkor a metódus az `Ures album!` szöveggel térjen vissza! (2 pont)

Példa használat:

```
const album = new Album('Call of the Wild');
album.cim = 'The Sacrament of Sin';
console.log(album.cim);                // 'The Sacrament of Sin'

album.zenetFelvesz({zeneCim: 'Incense & Iron', zeneHossz: 238});
album.zenetFelvesz({zeneCim: 'Killers With The Cross', zeneHossz: 250});
album.zenetFelvesz({zeneCim: 'Fire & Forgive'});

console.log(album.info());             // 'The Sacrament of Sin album, 2 darab zenevel'
console.log(album.albumHossz());       // '8 perc, 8 masodperc'
console.log(album.randomZene());       // 'Incense & Iron' VAGY 'Killers With The Cross'

const másikAlbum = new Album('The Sacrament of Sin');
masikAlbum.hossz = 488;
masikAlbum.zenek = ['Incense & Iron', 'Killers With The Cross'];
console.log(album.osszehasonlit(masikAlbum)); // true
```

Jó munkát!