

Test Subjects

Kuryák Dániel

Készítse el a leírás és a minta alapján az Aperture Science számára készült weboldalt, ahol a tesztalanyok adatait lehet megjeleníteni.

Megoldását a `main.js` fájlba írja! A tesztalanyok adatait tartalmazó `subjects` tömb a `data.js` fájlban található, ami elérhető a `main.js` fájlban is.

Tulajdonság	Típus	Leírás
<code>id</code>	Egész	A tesztalany azonosítója
<code>name</code>	Szöveg	A tesztalany neve
<code>birth</code>	Szöveg	A tesztalany születési dátuma (<code>null</code> ha nem ismert)
<code>occupation</code>	Szöveg	A tesztalany foglalkozása
<code>traits</code>	Tömb	A tesztalany tulajdonságai
<code>course</code>	Szöveg	A tesztalany által teljesített kurzus
<code>status</code>	Szöveg	A tesztalany állapota (<code>alive/terminated</code>)

1. Készítsen egy `userProfile` nevű objektumot, amelynek `name` tulajdonsága legyen az Ön neve (Keresztnév, illetve `position` tulajdonsága legyen egy, az Ön számára tetszőleges munkakör!
2. Állítsa be, hogy az előző feladatban elkészített objektum segítségével a `profile--name` azonosítójú elem szövege legyen az Ön neve, valamint a `profile--position` azonosítójú elem szövege legyen az Ön munaköre!
3. Egészítse ki a `createTableRow(subject)` függvényt az alábbi leírás alapján:
 - a. A `row` változóban eltárolt táblázat sora kapja meg a paraméterben megadott tesztalany azonosítóját `data` attribútumként!
 - b. Az `id` változóban eltárolt cella szövege legyen a tesztalany azonosítója!
 - c. A `name` változóban eltárolt cella szövege legyen a tesztalany neve!
 - d. A `status` változóban eltárolt `span` elem szövege legyen a tesztalany állapota! Használja a már elkészített `capitalize(text)` függvényt!
 - e. A `status` változóban eltárolt `spam` elem továbbá kapja meg a tesztalany státuszát, mint osztály!
4. Egészítse ki a `showSubject(subject)` függvényt úgy, hogy menjen végig a `subject` objektum kulcsain, majd végezze el az alábbi műveleteket:
 - a. Mentse az adott kulcsnak tartozó elemet egy változóba! Az elem azonosítója a következő: `current-subject--key`, ahol `key` az adott kulcs értéke!
 - b. Abban az esetben, ha a kulcs értéke “`status`” (azaz a tesztalany státuszár írjuk ki), akkor:
 - az elem szövege az objektum kulcsához tartozó érték úgy, hogy a `capitalize()` függvény segítségével nagybetűs legyen;
 - törölje le az elemről az `alive` és `terminated` osztályokat;
 - adja hozzá az objektum kulcsához tartozó érték a elemhez, mint osztály!
 - c. Abban az esetben, ha a kulcs értéke “`traits`” (azaz a tesztalany tulajdonságait írjuk ki), akkor:
 - hozzon létre egy tömböt;
 - menjen végig az objektum kulcsához tartozó értékeken egy ciklussal;
 - hozzon létre egy `span` elemet az összes tulajdonsághoz, aminek szövege legyen az adott tulajdonság, majd a `span` elemet adja hozzá a tömbhöz;
 - cserélje ki az elem gyermekét a tömb elemeire!
 - d. Abban az esetben, ha bármi más az értéke a kulcsnak, úgy az elem szövege legyen az objektum kulcsához tartozó érték!

5. Egészítse ki az `addEventListenersToRows()` függvényben lévő `find()` függvényt úgy, hogy az találja meg azt az elemet a tömbből, ahol az éppen vizsgált sor `id` data attribútuma megegyezik a tesztalany azonosítójával!
6. Egészítse ki a `filterSubjects(name)` függvényt úgy, hogy szűrje le a `subjects` tömböt aszerint, hogy a tesztalany neve tartalmazza-e a `name` paraméter értékét! Oldja meg, hogy ne legyen kis- és nagybetű érzékeny!