

Webtechnológiák beadandó

Szabó Bálint

Neptunkód: EJX162

Programtervező informatikus

kezdolap.html



A weboldal felső sávjában található egy menüsor, ennek a html és a css kódja mindegyik html-ben megegyezik. Ezen az oldalon csak egy üdvözlő szöveg és egy kép beszáras van (src="img/img.jpg").

```
<> kezdolap.html > html > body
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <style>
5  |   body {margin:0;}
6  |   </style>
7  |   <meta charset="utf-8">
8  |   <title>Szabó Bálint webtech</title>
9  |
10 |   <link rel="stylesheet" href="css/kezdolap.css">
11 |   </head>
12 |   <body>
13 |   <ul id="menu">
14 |   |   <li><a class="active" href="kezdolap.html">Kezdőlap</a></li>
15 |   |   <li><a href="leiras.html">Leírás</a></li>
16 |   |   <li><a href="jatek.html">Játék</a></li>
17 |   |   <li><a href="eredmenyek.html">Eredmények</a></li>
18 |   |   <li><a href="adatok.html">Adatok</a></li>
19 |   </ul>
20 |
21 |   <h1>Üdvözöllek a weblapomon</h1>
22 |   <h2>Egy logikai játékot fogok bemutatni</h2>
23 |   
24 |
25 |   </body>
26 </html>
```

kezdolap.css

A menüt egy listát használva hoztam létre. A menüt az id-ra és a listára hivatkozva formáztam meg a css fájlban.

```
1  html {
2      background-image:
3          linear-gradient(45deg, #008080, #FFA500);
4      background-repeat: no-repeat;
5      margin: 0;
6      width: 100%;
7      height: 100%;
8      font-size: large;
9  }
10
11  #menu {
12      width: 910px;
13      margin: 0 auto;
14      list-style: none;
15  }
16
17  #menu li {
18      float: left;
19  }
20
21  #menu a {
22      display: block;
23      text-align: center;
24      width: 150px;
25      text-decoration: none;
26  }
```

```
27  ul {
28      list-style-type: none;
29      margin: 0;
30      padding: 0;
31      overflow: hidden;
32      background-color: #222;
33      top: 0;
34      width: 100%;
35      font-size: x-large;
36  }
37
38  li {
39      float: left;
40  }
41
42  li a {
43      display: block;
44      color: white;
45      width: 100px;
46      text-align: center;
47      padding: 14px 16px;
48      text-decoration: none;
49  }
50
51  .active {
52      background-color: #CC5500;
53  }
```

leiras.html

Ezen a html oldalon használtam a pontozott listát, a táblázat létrehozását és formázását. Emellett a p, span, div és címsorok elemeket használtam.

A leiras.css-ben ilyen módon formáztam a táblázatot. A pozícióját is meghatároztam, hogy a szöveg mellett jobb oldalt helyezkedjen el.

```
table, tr, td {
    border: 1px solid black;
    border-collapse: collapse;
    width: 15%;
    height: 20%;
    text-align: center;
    font-size: 150%;
}

.tableposition {
    position: absolute;
    top: 120px;
    right: 350px;
    background-color: white;
}
```

```

20     <h2>Adatok:</h2>
21     <p>Itt tudod megadni az <span style="color:blue;font-weight:bold">adataidat</span>
22     , ami alapján látni fogod az eredményeidet.</p>
23     <h2>Játék:</h2>
24     <ul id="list">
25         <li>Itt tudod elindítani a játékot.</li>
26         <li>Egy csúszka segítségével ki tudod választani a
27         <span style="color:blue;font-weight:bold">nehézségi szintet</span>.</li>
28         <li>Egy stopper fogja mérni, hogy mennyi idő alatt oldod meg a játékot.</li>
29         <li>Látni fogod mennyit hibáztál</li>
30     </ul> </p>
31     <h2>Eredmények:</h2>
32     <p>Itt tudod megtekinteni a lejátszott játékaid eredményeit.</p>
33     <h2>Szabályok</h2>
34     <p>A kijelölt mezők számának <span style="color:blue;font-weight:bold">meg kell egyeznie
35     </span> a sorokba és az oszlopokban lévő számokkal.</p>
36     <table class="tableposition">
37         <caption>A játék egy lehetséges megoldása:</caption>
38         <tr>
39             <td></td>
40             <td>3</td>
41             <td>2</td>
42             <td>0</td>
43         </tr>
44         <tr>
45             <td>1</td>
46             <td bgcolor="green"></td>
47             <td></td>
48             <td></td>
49         </tr>
50         <tr>
51             <td>2</td>
52             <td bgcolor="green"></td>
53             <td bgcolor="green"></td>
54             <td></td>
55         </tr>
56         <tr>
57             <td>2</td>
58             <td bgcolor="green"></td>
59             <td bgcolor="green"></td>
60             <td></td>
61         </tr>
62     </table>

```

adatok.html

Ez a html oldal bekéri adatokat kér be a felhasználótól. Többféle űrlapot használva valósítottam meg. Az űrlapokban megadott adatokat a **localStorage**-ban eltároltam és az **eredmények.html** oldalon jelenítettem meg a **loadData()** metódussal. Ha a felhasználó új adatok ad meg, akkor a **localStorage.clear()**, függvény meghívása után az új adatokat fogja megjeleníteni.

forms.js

```
1  function getInputValue(){
2      localStorage.clear();
3      console.log(document.getElementById("fname").value);
4      localStorage.setItem("fname",document.getElementById("fname").value);
5      localStorage.setItem("lname",document.getElementById("lname").value);
6      localStorage.setItem("username",document.getElementById("username").value);
7      localStorage.setItem("inputDate",document.getElementById("inputDate").value);
8      localStorage.setItem("colors",document.getElementById("colors").value);
9
10     if(document.getElementById("cbNapi").checked)
11     {
12         localStorage.setItem("gtime",document.getElementById("cbNapi").value);
13     } else if(document.getElementById("cbHeti").checked)
14     {
15         localStorage.setItem("gtime",document.getElementById("cbHeti").value);
16     } else if(document.getElementById("cbHavi").checked)
17     {
18         localStorage.setItem("gtime",document.getElementById("cbHavi").value);
19     }
20
21     console.log("adatok mentve");
22 }
23
24 function colorData(){
25     var ids = ['output_fname','output_lname','output_username','output_inputDate','output_gtime'];
26
27     ids.forEach(id => {
28         document.getElementById(id).style.color = localStorage.getItem("colors");
29     });
30 }
```

A **forms.js**-ben létrehoztam egy validáló függvényt. Ennek az a szerepe, hogy ha a felhasználó nem tölti ki a Vezetéknév vagy Születési dátum forms-ot, akkor egy alert-et kap a kötelező kitöltésről. Ezt követően az űrlap mező kerete pirosra változik ezzel is jelezve a hibát.

```
41  ✓ function validateForm() {
42      let x = document.forms["names"]["fname"].value;
43      let z = document.getElementById("inputDate").value;
44  ✓  if (x == "") {
45      document.getElementById("fname").style.borderColor = "red";
46      alert("Name must be filled out");
47      return false;
48  ✓  } else if (document.getElementById("inputDate").value == "") {
49      document.getElementById("inputDate").style.borderColor = "red";
50      alert("Date must be filled out");
51      return false;
52  }
```

jatek.html, table.js

Ezen az oldalon, létrehoztam egy slider-t, ami 3 értéket tud felvenni. A table.js-ben az onSlideChange() függvényben, megvizsgálja a slidet aktuális értékét, és ez által generál le egy táblázatot. A slider értékét változtatva változik a táblázat mérete. A jatek.css-ben formáztam a táblázatot a divStyle segítségével (az appendChild() segítségével hozzáadtam táblázatot a div-hez).

```
1  function onSliderChange(){
2      var size;
3      if(document.getElementById("tableSlider").value == 1){
4          size = 4;
5      } else if (document.getElementById("tableSlider").value == 2){
6          size = 5;
7      } else {
8          size = 6;
9      }
10     document.getElementById("divTable").remove();
11     var table = document.createElement("table");
12     table.setAttribute("id", "main");
13     for(var i = 1; i<=size ;i++){
14         var tr = document.createElement("tr");
15         for(var j = 1; j<=size ;j++){
16             var td1 = document.createElement("td");
17             if(i%2 == 0){
18                 var text1 = document.createTextNode("X");
19             }else {
20                 var text1 = document.createTextNode("O");
21             }
22             td1.appendChild(text1);
23             tr.appendChild(td1);
24         }
25         table.appendChild(tr);
26     }
27
28     var div = document.createElement("div");
29     div.appendChild(table);
30     div.classList.add("divStyle");
31     div.setAttribute("id", "divTable");
32     document.body.appendChild(div);
33 }
```

```
35  function onSliderLoad(){
36      var size = 5;
37      var table = document.createElement("table");
38      table.setAttribute("id", "main");
39      for(var i = 1; i<=size ;i++){
40          var tr = document.createElement("tr");
41          for(var j = 1; j<=size ;j++){
42              var td = document.createElement("td");
43              if(i%2 == 0){
44                  var text = document.createTextNode("X");
45              }else {
46                  var text = document.createTextNode("O");
47              }
48              td.appendChild(text);
49              tr.appendChild(td);
50          }
51          table.appendChild(tr);
52      }
53      var div = document.createElement("div");
54      div.appendChild(table);
55      div.classList.add("divStyle");
56      div.setAttribute("id", "divTable");
57      document.body.appendChild(div);
58  }
```