



Ondokuz Mayıs Üniversitesi, Web Programlama Labı

Ad - Soyad: Alper Karaca

Öğrenci Numarası: 21060619

Ödev Adı: Föy - 2

[GitHub Linki](#)

Özet Bilgi

Bu föy içerisinde AJAX kullanarak bir XML dosyasından öğrenci bilgileri tablo içerisinde yazdırılır ardından seçilen öğrencinin detayları belirtilir.

Kaynak Kodları ve Açıklamaları

Ad Soyad	Numara
Alper Karaca	11111111
Ali Veli	22222222
Doğukan Ceylan	33333333
Hüseyin Aşan	44444444

Ad - Soyad: Doğukan Ceylan
Numara: 33333333
Yıl: 4
Bölüm: İnşaat Mühendisliği

Ad Soyad	Numara
Alper Karaca	11111111
Ali Veli	22222222
Doğukan Ceylan	33333333
Hüseyin Aşan	44444444

```
1 <students>
2
3   <student>
4     <name>Alper Karaca</name>
5     <number>11111111</number>
6     <year>2</year>
7     <major> Bilgisayar Mühendisliği</major>
8   </student>
9
10  <student>
11    <name>Ali Veli</name>
12    <number>22222222</number>
13    <year>3</year>
14    <major> Elektrik Elektronik Mühendisliği</major>
15  </student>
16
17  <student>
18    <name>Doğukan Ceylan</name>
19    <number>33333333</number>
20    <year>4</year>
21    <major> İnşaat Mühendisliği</major>
22  </student>
23
24  <student>
25    <name>Hüseyin Aşan</name>
26    <number>44444444</number>
27    <year>4</year>
28    <major> Endüstri Mühendisliği</major>
29  </student>
30
31 </students>
```

Öğrenci verilenin tutulduğu XML dosyasının içeriği bu şekildedir. Student tagları arasında öğrenci verileri tutulmaktadır.

```
1 <!DOCTYPE html>
2 <html lang="tr">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <title>AJAX Alıştırması</title>
7     <link rel="stylesheet" href="css/style.css" />
8   </head>
9   <body>
10
11     <div id="student-det"></div>
12
13     <div id="students-table-container">
14       <table id="students-table">
15         <tr>
16           <td><strong>Ad Soyad</strong></td>
17           <td><strong>Numara</strong></td>
18         </tr>
19         <tbody id="table-body"></tbody>
20       </table>
21     </div>
22
23     <script src="scripts/app.js"></script>
24   </body>
25 </html>
26
```

Sayfanın HTML kodu yukarıdaki gibidir. Tablo içerisinde statik olarak gösterilecek veriler yer almaktadır. Sayfa yüklendiği zaman tablo XML dosyasındaki veriler ile doldurulur.

```
1 function loadDoc() {
2     var xhttp = new XMLHttpRequest();
3     xhttp.onreadystatechange = function () {
4         if (this.readyState == 4 && this.status == 200) {
5             listStudents(this);
6         }
7     }
8     xhttp.open("GET", "../data/datalist.xml", true);
9     xhttp.send();
10 };
11
12 listStudents = (xml) => {
13     var xmlDoc = xml.responseXML;
14     const students = xmlDoc.getElementsByTagName("student");
15     let output = "";
16     for (let i = 0; i < students.length; i++) {
17         const student = students[i];
18         const name = student.getElementsByTagName("name")[0].childNodes[0].nodeValue;
19         const number = student.getElementsByTagName("number")[0].childNodes[0].nodeValue;
20         output += `<tr onclick="getStudentDetails(${i})"><td>${name}</td><td>${number}</td></tr>`;
21     }
22     document.getElementById("table-body").innerHTML = output;
23 }
24
25 window.onload = loadDoc;
```

Bu JavaScript dosyası içerisindeki loadDoc fonksiyonu sayfa yüklendiği zaman yapılacakları belirtmektedir.

Öncelikle bir XMLHttpRequest objesi oluşturulur ve durum değişikliği durumunda listStudents fonksiyonu çağrılır.

Objenin okuma yapacağı XML dosyasının yol bilgileri verilir ve okuması sağlanır işlemler tamamlandıktan sonra send() fonksiyonu ile sonuç gönderilir.

listStudents() fonksiyonu belirtilen XML dosyasını parse işlemini yapar.

Bu kod bloğunda verilen XML dosyasındaki student taglerine göre ayrıştırılır ve for döngüsü içerisinde tabloda gözükecek olan numara ve isim bilgileri ilgili değişkenlere atanır.

En sonunda Output değişkeninde oluşacak olan tablo satırları biçimlendirilir ve table-body ID'sine sahip içeriği değiştirilir.

Bu işlemin sayfa yüklendiği zaman yapılmasını sağlamak için de window.onload eventinde loadDoc fonksiyonu çağrılır.

```

28 getStudentDetails = (index) => {
29     var xhttp = new XMLHttpRequest();
30     xhttp.onreadystatechange = function () {
31         if (this.readyState == 4 && this.status == 200) {
32             showStudentDetails(this, index);
33         }
34     }
35     xhttp.open("GET", "../data/datalist.xml", true);
36     xhttp.send();
37 }
38
39 showStudentDetails = (xml, index) => {
40     var xmlDoc = xml.responseXML;
41     const students = xmlDoc.getElementsByTagName("student");
42     const student = students[index];
43     const name = student.getElementsByTagName("name")[0].childNodes[0].nodeValue;
44     const number = student.getElementsByTagName("number")[0].childNodes[0].nodeValue;
45     const year = student.getElementsByTagName("year")[0].childNodes[0].nodeValue;
46     const major = student.getElementsByTagName("major")[0].childNodes[0].nodeValue;
47
48     const output =
49     "<div id='container'>" + "<div id='student-details'>" +
50     `
51     <p>
52         <strong>Ad - Soyad:</strong> ${name}<br>
53         <strong>Numara:</strong> ${number}<br>
54         <strong>Yıl:</strong> ${year}<br>
55         <strong>Bölüm:</strong> ${major}<br>
56     </p>
57     ` +
58     "</div> </div>" ;
59     document.getElementById("student-det").innerHTML = output;
60 }

```

getStudentDetails fonksiyonu sayesinde üzerine tıklanan öğrencinin index numarası alınır ve bu sayede XML dosyası içerisinde seçme işlemini yapabiliriz.

showStudentDetails() fonksiyonu seçilen öğrencinin bilgilerini çeker ve bunları ilgili değişkenlere atar.

En sonunda oluşturulacak olan divler için burada container ve student-details ID'lerine sahip elemanlar oluşturulur ve içeriği doldurulur.

Oluşturulan bu HTML bloğu student-det ID'li HTML elemanı ile yer değiştirir.

```
1 @import url('https://fonts.googleapis.com/css2?
  family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,60
  0;1,700;1,800;1,900&display=swap');
2
3 body {
4   font-family: 'Poppins', sans-serif;
5   margin: 0;
6   padding: 0;
7   background-color: #24273a;
8   color: #181926;
9 }
10
11 #students-table-container {
12   display: flex;
13   align-content: center;
14   justify-content: center;
15 }
16
17 #students-table {
18   width: 50%;
19   border: 1px solid #000;
20   border-collapse: collapse;
21 }
22
23 #students-table td {
24   border: 1px solid #000;
25   padding: 5px;
26 }
27
28 #students-table tr:nth-child(even) {
29   background-color: #ed8796;
30   transition: all 0.3s;
31 }
```

Bu kodun CSS dosyasının içeriğidir. Öncelikle Google Fonts üzerinden Poppins fontunu ekledim ardından body içerisinde tüm fontu uyguladım ve arka planı rengini değiştirdim.

Tabloyu içerisinde tutan div için flex görünüm verip ortaladım.

Tablonun genişliğini ekranın yarısını kaplayacak şekilde düzenledim.

Tablodaki satırlara border ve iç dolgunluk verdim ardından teklik – çiftlik durumuna göre arka plan renklerini değiştirdim ve bu işlemler arasında hover olduğu durumlar için geçiş animasyonunu yumuşatmak adına 300mslik bir geçiş süresi ekledim.


```
37 #students-table tr:nth-child(odd) {
38     background-color: #ee99a0;
39     transition: all 0.3s;
40 }
41
42 #students-table tr:hover {
43     background-color: #f5a97f;
44     transition: all 0.3s;
45     cursor: pointer;
46 }
47
48 #container {
49     margin-left: 37.5%;
50     border-radius: 10px;
51     background-color: #a6e3a1;
52     width: 25%;
53 }
54
55 #student-det {
56     margin-top: 5rem;
57     margin-bottom: 5rem;
58 }
59
60 #student-details {
61     display: flex;
62     flex-direction: column;
63     align-items: center;
64     padding: 10px;
65     transition: all 1s;
66 }
```

Bu kısımda hover durumunda arka planın değişmesini ve imlecin işaret halini almasını sağladım.

Öğrenci verilerinin gösterileceği yerler için soldan %37.5'lik bir margin bırakıp 10px lik büyüklük verip dairesellik ekledim. Genişliğini %25 olarak ayarlayıp arka plan rengini değiştirdim.

student-det ID'sine sahip eleman için alt ve üstten 5remlik boşluk ekledim.

Öğrenci verilerinin gösterileceği iç bloğu ortalamak için flex yapısı kullandım ve 10pxlik dış dolgu verip daha dolgun gösterdim.