



# **Node.js ile Web Programlama**

## **Rapor - 8**

Mahire Zühal Özdemir

## İÇİNDEKİLER

ŞEKİLLER DİZİN .....	3
Partials .....	4
HBS.....	4
404 Page .....	7

## ŞEKİLLER DİZİNİ

Şekil 1:HBS Modülü	4
Şekil 2: Static Header	4
Şekil 3:Help.hbs Dosyası	5
Şekil 4: Help Sayfası	5
Şekil 5: Index.js Sayfası	6
Şekil 6: Dinamik Title	6
Şekil 7: Dinamik Title	6
Şekil 8: 404 Sayfası	7
Şekil 9: 404 Page	7

## Partials

Partials, başlık ve altbilgi gibi belirli bileşenlerin tekrar kullanılmasını sağlayan Handlebars.js özelliğidir. Bu, web uygulamalarında tutarlılık sağlamak için yaygın olarak kullanılır. Bu kısımların her birini ayrı ayrı her sayfada yeniden oluşturmak yerine, koddaki tekrarı azaltır ve bakımı kolaylaştırır.

## HBS

hbs, Handlebars.js şablon motorunu Node.js ortamına entegre eden bir modüldür. Handlebars.js, HTML şablonlarını daha dinamik hale getirmek için kullanılan bir JavaScript şablon motorudur.

Handlebars, basit ve temiz bir sözdizimine sahiptir ve bir HTML dosyası içinde JavaScript ifadeleriyle etkileşimli içerik oluşturmanıza izin verir. Özellikle web uygulamalarında, sunucu tarafında oluşturulan içeriğin istemci tarafında kullanıcıya sunulması gerektiğinde çok kullanışlıdır.

Hbs modülünü uygulamaya dahil etmek ve Handlebars.js şablon motorunu kullanmak için import işlemini gerçekleştirelim.

```
// npm install hbs
const hbs = require('hbs');
```

Şekil 1:HBS Modülü

Uygulama içerisinde aşağıdaki değişiklikler yapıldı.

- templates dizini içinde views ve partials adında iki alt dizin oluşturuldu.
- header.hbs dosyası oluşturuldu ve içeriği <h1>Static Header.hbs Text</h1> olarak belirlendi.

```
templates > partials > 📄 header.hbs > ...
1    <h1>Static Header.hbs Text</h1>
```

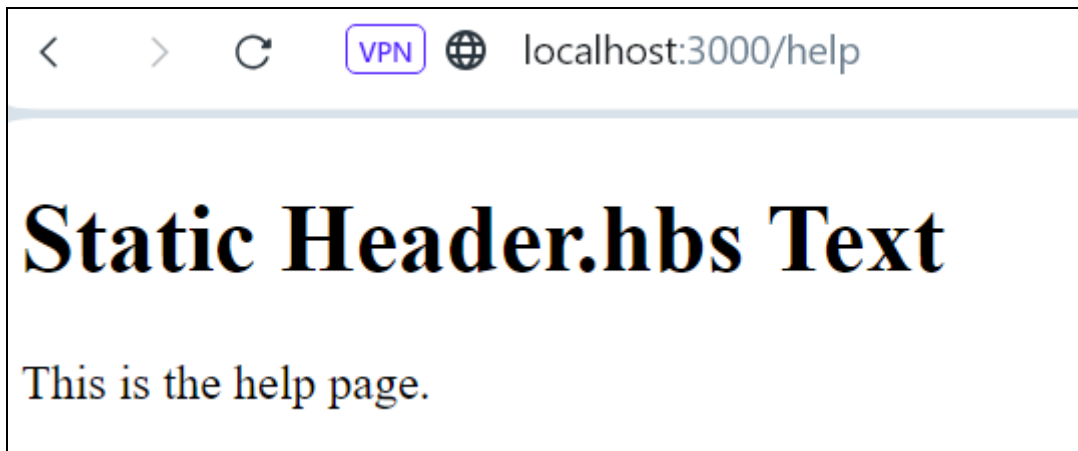
Şekil 2: Static Header

- help.hbs dosyası içeriği güncellendi ve {{>header}} ifadesi eklenerek başlık kısmına header parçası eklenmesi sağlandı.

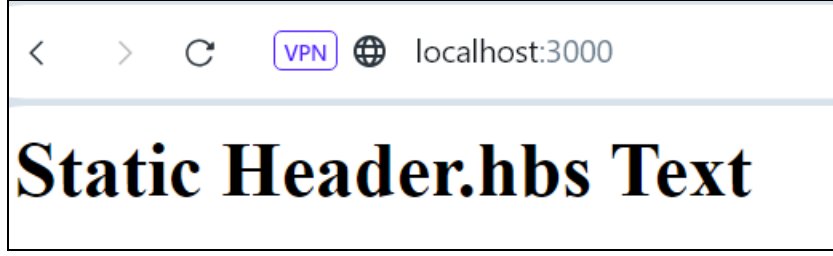
```
templates > views > help.hbs > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <link rel="stylesheet" href="/css/styles.css">
5  </head>
6  <body>
7  {{>header}}
8  <p>{{helpText}}</p>
9  </body>
10 </html>
```

Şekil 3: Help.hbs Dosyası

- index.hbs ve about.hbs dosyalarının içeriği güncellendi ve her ikisine de {{>header}} ifadesi eklendi.
- Nodemon ile app.js dosyası başlatıldı, bu sayede sunucu otomatik olarak yeniden başlatıldı.
- Tarayıcıda <http://localhost:3000/help>, <http://localhost:3000/about> ve <http://localhost:3000/> adresleri ziyaret edilerek sayfaların kontrol edilmesi sağlandı.



Şekil 4: Help Sayfası



Şekil 5: Index.js Sayfası

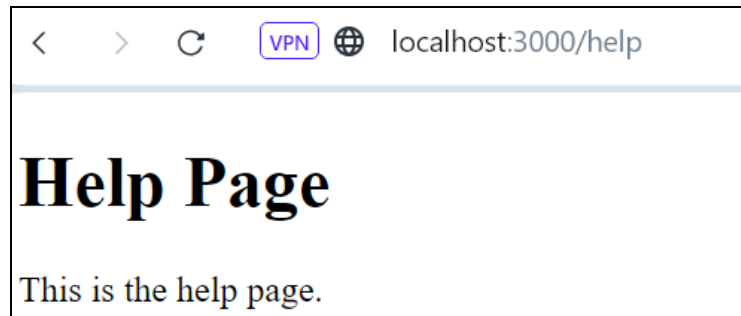
**App.js:** app.get fonksiyonu içindeki /help rotası için res.render fonksiyonunun içine title ve name değişkenleri ekleniyor. Bu değişkenler, help.hbs dosyasında kullanılmak üzere gönderiliyor.

**header.hbs:** Statik başlık yerine, {{title}} değişkenini içeren bir başlık ekleniyor. Bu değişken, app.js dosyasından res.render fonksiyonu aracılığıyla help.hbs dosyasına iletilen başlık bilgisini temsil ediyor.

Bu değişikliklerle, app.js dosyası artık /help rotasına yapılan isteklerde title ve name değişkenlerini içeren help.hbs şablonunu render etmekte. header.hbs dosyasında ise artık statik bir başlık yerine, title değişkeni kullanılmakta. Bu sayede help.hbs dosyasına iletilen title ve name değişkenleri başlık kısmında dinamik olarak gösterilebilmekte.

```
app.get('/help', (req, res) => {  
  res.render('help', {  
    helpText: 'This is the help page.',  
    title: 'Help Page', // Yeni eklenen başlık değişkeni  
    name: 'Cem Can' // Yeni eklenen isim değişkeni  
  });  
});
```

Şekil 6: Dinamik Title



Şekil 7: Dinamik Title

## 404 Page

Express.js'de 404 sayfasını oluşturmak için, app.js dosyasında, diğer rotalardan sonra ek bir rotaya ihtiyacımız olacak. Bu rota, hiçbir diğer rotaya uymayan istekleri yakalayacak ve 404 sayfasını gösterecek. Örnek bir 404 sayfası rotası şu şekilde olabilir:

```
// 404 sayfası rotası
app.get('*', (req, res) => {
  res.status(404).render('404', {
    title: '404 Not Found',
    errorMessage: 'Sayfa bulunamadı.'
  });
});
```

Şekil 8: 404 Sayfası

404 sayfasının görüntüsünü oluşturmak için, views dizini içinde 404.hbs adında bir Handlebars dosyası oluşturmamız gerekecek. Bu dosyada, 404 hatası için uygun bir görsel veya mesaj yer alabilir. Örneğin, 404.hbs dosyası şu şekilde olabilir:

```
templates > views > 404.hbs > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <link rel="stylesheet" href="/css/styles.css">
5  </head>
6  <body>
7  {{>header}}
8  <p>{{errorMessage}}</p>
9  {{>footer}}
10 </body>
11 </html>
```

Şekil 9: 404 Page

Oluşturulan sayfalar CSS yardımı ile şekillendirilebilir.