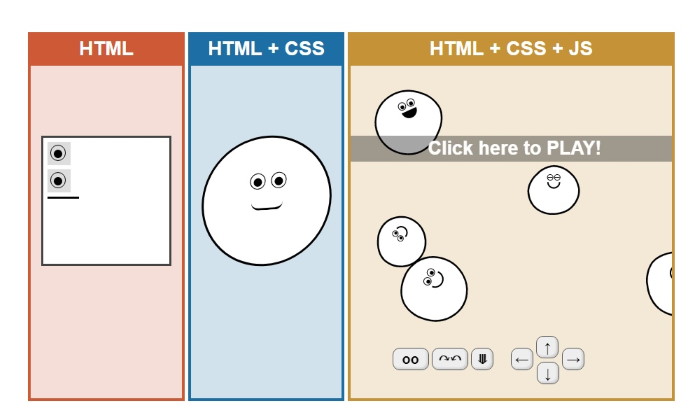
DERS NOTLARI VE TEKRAR TAKİBİ

Days\_47: JavaScript - jquery konu anlatımı \*\*FrontEnd/Days\_6-7 ---> 15/06/2025



# JavaScript nedir?

Geliştiriciler tarafından etkileşimli web sayfaları oluşturmak için kullanılır. Bu dil sunucu tarafı programlama, mobil uygulamalar, oyunlar ve masaüstü uygulamaları için kullanılabilir…

🡪Selector (seçici): HTML belgesi içindeki belirli bir elementi veya element grubunu seçmek için kullanılan tanımlayıcıdır. CSS ve JavaScript bu seçicileri kullanarak HTML öğelerine ulaşır.

CSS ve JavaScript içinde selector/seçici kavramı: HTML öğeleri üzerinde stil (CSS) veya işlem (JavaScript) uygulamak için bu "seçiciler" kullanılır.

\*\*CSS'de selektörler, belirli HTML öğelerine stil vermek için kullanılır.

Örn.

| **Selector Türü** | **Açıklama** | **Örnek** |
| --- | --- | --- |
| **Element selector** | HTML tag’ına göre seçer | p { color: red; } |
| **Class selector** | class değerine göre | .kirmizi { color: red; } |

\*\* JavaScript içinde DOM (Document Object Model) üzerinden HTML öğelerini seçmek için kullanılır.

Örn.

let paragraf = document.getElementById("mesaj");

Eğer seçici kullanmazsak bu objeye karşı tarafın hakimiyetini bekleyemeyiz. Mesela instagramda beğenme butonu yani kalp başta beyaz tıklıyoruz arka plan kırmızı oluyor sağlıyoruz. İd … olan bir like butonum var o like butonuna git ve arka planını kırmızı renge çevir… işte bunu seçici ile yapıyoruz. Seçicilerde,

document.getElementById("mesaj"); ile çalışır.

Document: js çalıştırıldığı ve o anda derlendiği yer, sahne

getElementById: documentin içine bak ve id şu olan veriyi getir.

🡪Const citiesObj = document. getElementById(‘cities’); ifadesi bellekte bir maliyet oluştutur mu?

Hayır, Bellekte zaten cities elemanı var biz cities’e referansını cities objesine atanamasını sağlıyoruz.

Object: Programlamada nesneler, birden çok veriyi ve davranışı (fonksiyonları) tek bir yapıda birleştirmemizi sağlar. JavaScript’te obje (object), birden fazla özellik (property) ve fonksiyon (method) içerebilen veri tipidir.

**Örn.**

function nesne = {

anahtar1: deger1,

key: value,

};

- Obje, anahtar-değer (key-value) çiftlerinden oluşur.

- Her değer, ister veri ister fonksiyon (method) olabilir.

-Obje, gerçek dünyadaki varlıkları yazılımda temsil etmek için çok kullanılır.

**Örn.**

let araba = {

marka: "Toyota",

model: "CVR",

yil: 2025,

calistir: function() {

console.log("Araba çalıştı!");

}

};

\*\*arr.push(obj) // dizi içine veri ekleme

\*\*Buraya kadar olan notların kod dökümünün dosya yolu 🡪 belgeler/github/workspace/frontend/Days\_6

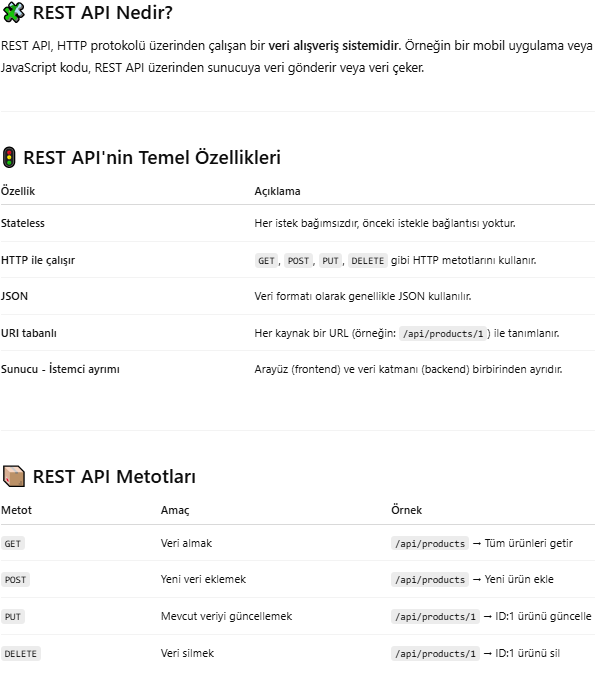
----------------------------------------------------------- \*\*\* ----------------------------------------------------------

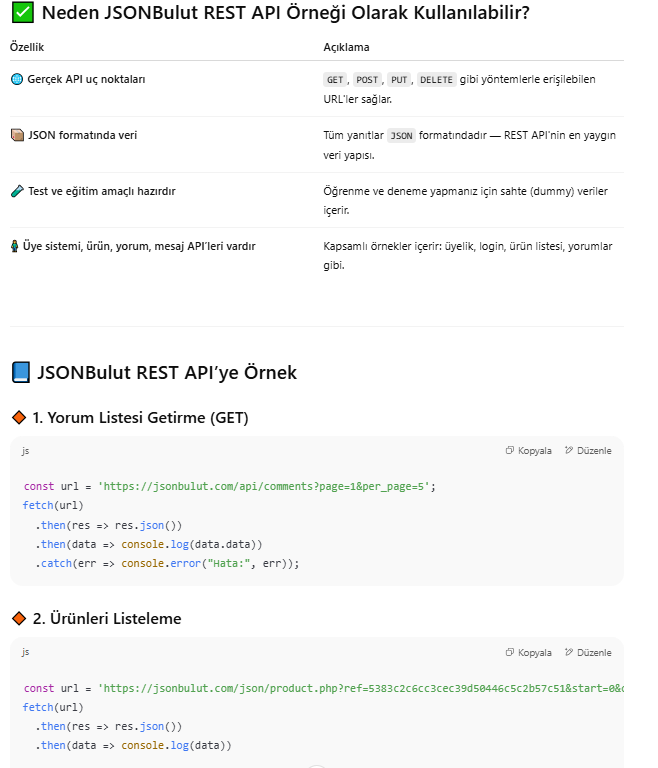
Bu kısımda ki notların kod dökümünün dosya yolu 🡪 belgeler/github/workspace/frontend/Days\_7

AJAX (Asynchronous JavaScript and XML), JavaScript kullanarak sayfa yeniden yüklenmeden web sunucusuyla veri alışverişi yapmamızı sağlayan bir tekniktir.

🧠 Temel Tanım:

* AJAX, web sayfasının arka planda veri alıp göndermesini sağlar.
* Sayfanın yalnızca bir kısmını güncelleyerek kullanıcı deneyimini geliştirir.
* Genellikle API'lerle (REST API) çalışmak için kullanılır.





Yukarıdaki gibi örnek uç noktalar REST API yapısına birebir uyar:

* Veri çekme: GET
* Yeni veri gönderme: POST
* Veri güncelleme: PUT
* Veri silme: DELETE

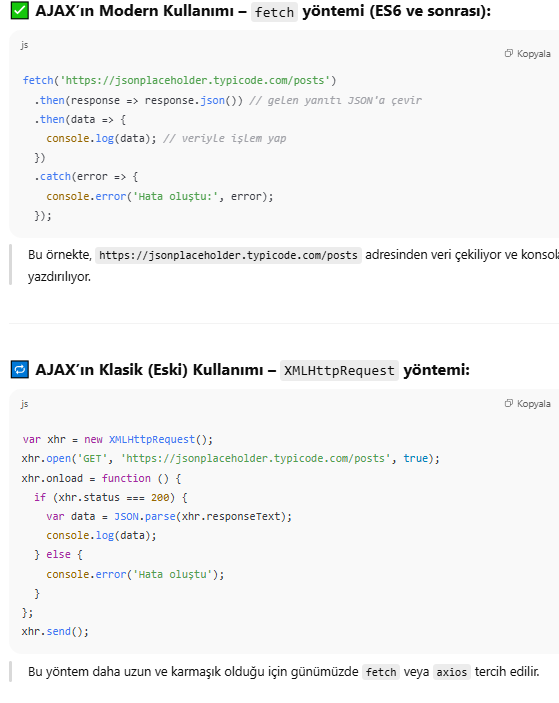
**🔐 Not:**

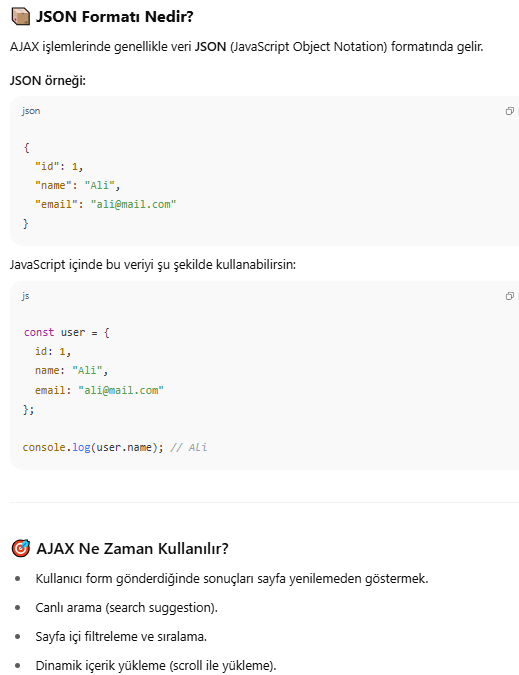
Bazı API'ler ref yani referans anahtarı (API key) gerektirir. JSONBulut da bu yöntemi kullanır. Ücretsiz üye olarak kendi ref anahtarını alıp testler yapabilirsin.

**🔧 Sonuç**

Evet, jsonbulut.com sitesi:

* REST API mantığını anlamak,
* JavaScript ile AJAX/Fetch uygulamaları yapmak,
* Projelerinde sahte veri kullanarak test gerçekleştirmek için **mükemmel bir örnek platformdur.**





\*\*\*İki ters tek tırnak gibi görünen şu işaret: ` (klavye üzerinde genellikle ESC tuşunun altında, 1 tuşunun

Backtick işareti, yazılım dünyasında farklı programlama dillerinde çeşitli amaçlarla kullanılır:

| **Dil / Ortam** | **Kullanımı** |
| --- | --- |
| JavaScript/TypeScript | Template literals, ${} ile değişken gömme |
| Markdown | Kod blokları ve satır içi kod |
| Shell (Bash) | Komut çıktısı almak (eski yöntem) |
| MySQL | Tablo/sütun adlarını belirtmek |
| Python 2 | repr() işlevi (artık önerilmez) |

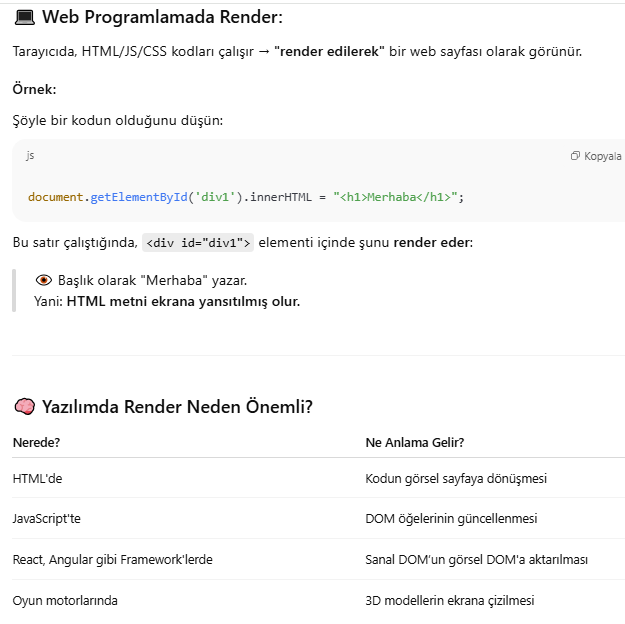
****

****

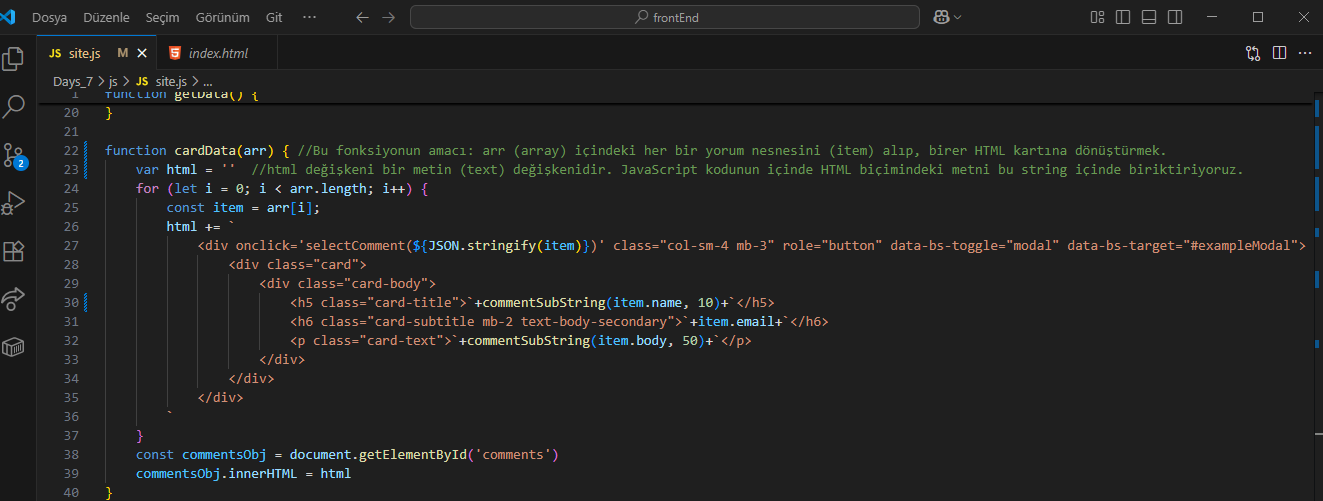
\*\*\* Render, kelime anlamı olarak:

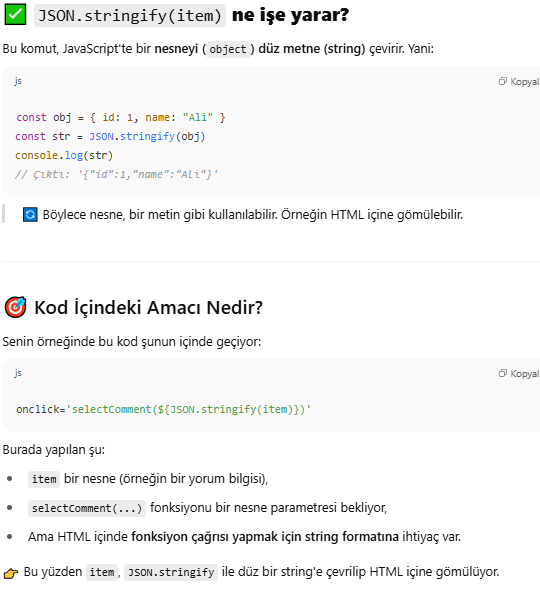
“Görselleştirmek” ya da “ekrana çizmek/göstermek” anlamına gelir.

Yazılımda ise: Veri veya kodun, kullanıcıya görsel olarak sunulmasıdır.



\*\*\*Kod bloğunun açıklaması:







Bu kısma kadar olan notların kod dökümünde de detaylı not alımı var bkz. Days\_7/site.js

----------------------------------------------------------- \*\*\* ----------------------------------------------------------

Days\_48: jquery - JavaScript’i daha az kod yazarak nasıl yönetiriz? \*\*FrontEnd/Days\_8---> 21/06/2025

### **jQuery**, JavaScript dilinde yazılmış, açık kaynaklı ve hafif bir **JavaScript kütüphanesidir**. Web sayfalarında JavaScript kullanımıyla yapılabilecek işleri **daha kısa, daha kolay ve tarayıcı uyumlu şekilde yapmanı** sağlar.

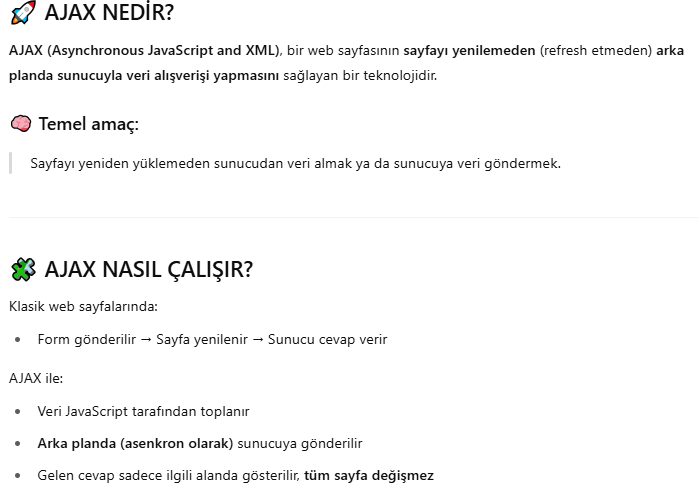
### 

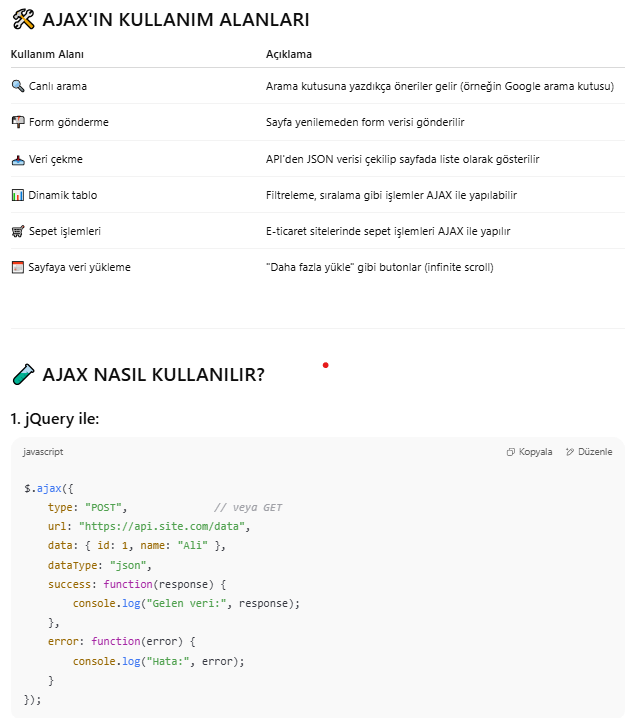
### 

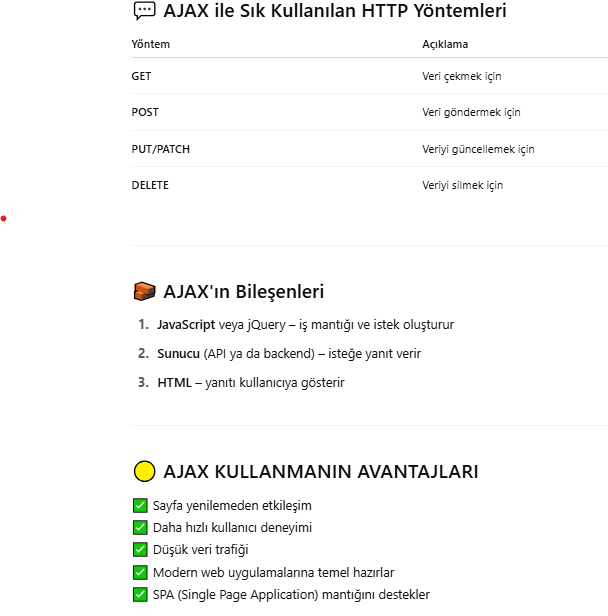
\*\*\*Bkz. bu dersin ekser notları FrontEnd/Days\_8 klasöründe tekrar et!

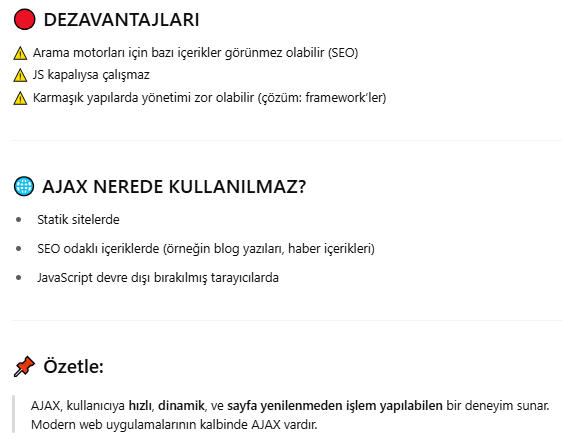
Days\_49: jquery - JavaScript - AJAX kullanımı \*\*FrontEnd/Days\_9---> 22/06/2025

\*\*bu ders, Days\_48 de işlenen jquery ve js konularının tekrarı yapıldı ve ajax konusu işlendi

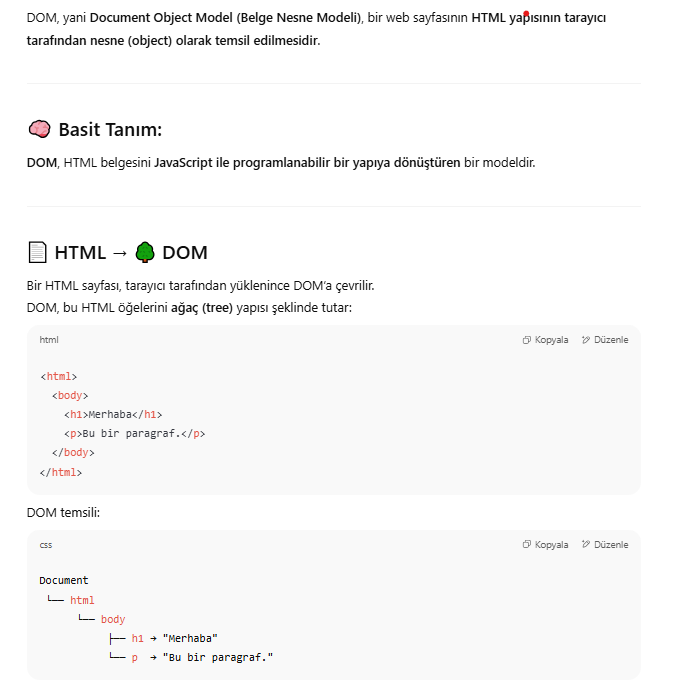


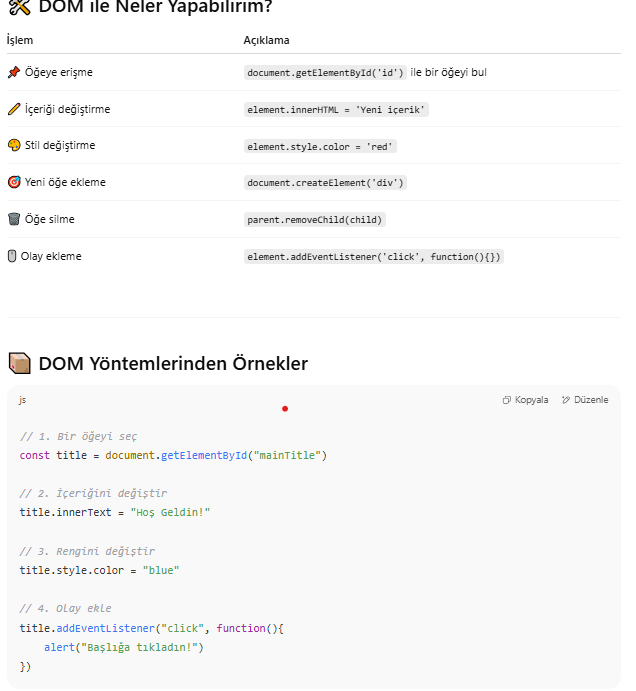


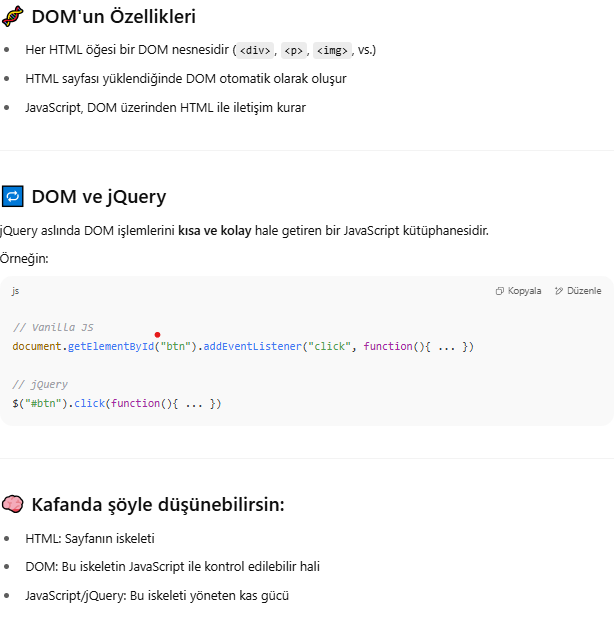




### DOM NEDİR? JavaScript ya da jQuery ile **sayfadaki öğelere erişmek**, **değiştirmek**, **silmek** ya da **yeni öğeler eklemek** için DOM kullanılır.







\*\*\*jsonbulut.com site tanıtımı 🡪 <https://jsonbulut.com/>

jsonbulut.com, **Free Fake REST API** hizmeti sunan bir platformdur. Bu servis, geliştiricilerin gerçek bir sunucu kurmadan web ve mobil uygulamalar için REST API çağrılarını hızlıca test edip prototip oluşturmalarına olanak tanır.

Temelde, jsonbulut.com kullanıcılarına yorumlar (comments), gönderiler (posts), ürünler (products), yapılacaklar (todos) ve kullanıcılar (users) gibi sık kullanılan verilerle aşağıdaki işlemleri deneme fırsatı sağlar:

* **GET, POST, PUT, DELETE** talepleriyle API uç noktalarının nasıl çalıştığını görselleştirme
* JWT (JSON Web Token) ile kimlik doğrulama kullanıcı akışı test etme
* Veriler üzerinde silme, güncelleme, listeleme gibi operasyonlar gerçekleştirme [jsonbulut.com+1jsonbulut.com+1](https://jsonbulut.com/apidocs?utm_source=chatgpt.com)

Geliştirme sürecinizde kendi backend’inizi yazmak yerine bu hazır sahte API’yi kullanarak ön yüz (frontend) ya da mobil uygulamanızı hızla geliştirebilir, test edebilirsiniz. Giriş yapmanız gereken senaryolar için örnek kullanıcı bilgileri de (e‑posta/parola) sağlanmış durumda .

**Özetle**

| **Konu** | **Açıklama** |
| --- | --- |
| Amaç | API isteklerini hızlıca test etmek, prototip geliştirmek |
| Sağlananlar | yorumlar, gönderiler, kullanıcılar, ürünler, yapılacaklar gibi uç noktalar |
| İşlemler | GET, POST, PUT, DELETE, JWT destekli authentication |
|  |  |

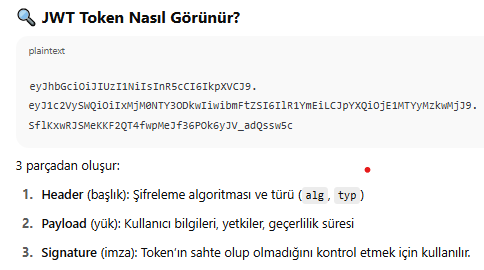
### 🔐 JWT Token Nedir? Bkz.-->Days\_9/login.js/39.kod satırı

JWT Token (JSON Web Token), bir kullanıcının kimliğini doğrulamak ve bu kimliğe bağlı yetkileri taşımak için kullanılan **şifreli bir dijital kimlik kartıdır**. Web uygulamalarında oturum (login) yönetimi için sıkça kullanılır.

JWT, kullanıcı giriş yaptıktan sonra sunucunun oluşturup istemciye (örneğin tarayıcıya) verdiği **bir metin yapısıdır**. Bu token:

* Kullanıcının **kim olduğunu**,
* Hangi işlemleri yapabileceğini (**yetki**),
* Ne kadar süreyle geçerli olduğunu belirtir.

Bu token genellikle her istekte sunucuya gönderilir, böylece sunucu tekrar tekrar kullanıcıdan kullanıcı adı/parola istemez.



**✅ JWT Ne İşe Yarar?**

* Kullanıcı giriş yaptıktan sonra access\_token sunucu tarafından döner.
* Tarayıcı bu token’ı localStorage/sessionStorage gibi bir yerde saklar.
* Kullanıcı her API isteğinde bu token’ı sunucuya gönderir.
* Sunucu, bu token’ı kontrol edip kullanıcının gerçekten giriş yapmış ve yetkili olup olmadığını anlar.

**🧠 Örnek Senaryo:**

1. Kullanıcı giriş yapar.
2. Sunucu access\_token yani JWT token döner:

js

KopyalaDüzenle

response.data.access\_token = "eyJhbGciOi..."

1. Tarayıcı bu token’ı saklar.
2. Artık kullanıcı sayfalar arasında dolaşırken tekrar giriş yapmasına gerek kalmaz.
3. Token süresi bittiğinde yeniden giriş yapılması gerekir.

\*\*\* <https://jsonviewer.stack.hu/> sitesi, JSON (JavaScript Object Notation) formatındaki verileri **okunabilir bir şekilde görüntülemek ve analiz etmek** için kullanılan **ücretsiz bir çevrimiçi araçtır**.

**📌 Ne zaman kullanılır?**

* API'den gelen karmaşık JSON cevabını **okunur hale getirmek için**
* JWT Token içindeki base64 çözümlenmiş payload'ı anlamak için
* JSON verisini debug (hata ayıklama) etmek için
* Kodlama yaparken test JSON verisi hazırlamak için