

HTTP

Bu yazımda sizlere internetin en eski protokollerinden birisi olan HTTP protokolünü anlatmaya çalışacağım. Aslında benim için HTTP protokolü, bildiğim en kötü protokollerin başını çeker, bunun böyle olmasının nedeni ise durumsuz bir protokol olması. Yani bir önceki HTTP request-response life cycle ile ilgili hiç bir şey hatırlamıyor oluşu, tabi ki de şunu da unutmamak gerekli, HTTP Protokolünün oluşturulma amacı bulunduğu yıl itibari ile belge transferi için kullanılan bir protokoldü, internetin gelişimi ve Word Wide Web teknolojisinin hayatımıza girmesi ile birlikte, HTTP protokolü ile çalışabilecek teknolojiler gelişti. Bunlara örnek vermek gerekirse cookie,token vb. ama Ben bu yazımda size HTTP protokolünün ilgili web sitesine istek atarken, nasıl bir işlem üzerinden geçtiğini lanse etmeye çalışacağım. yani herkesin bildiği request-response durumunu. Öncelikle HTTP Protokolünün Özelliklerini başlıklar halinde sıralamak gerekirse.

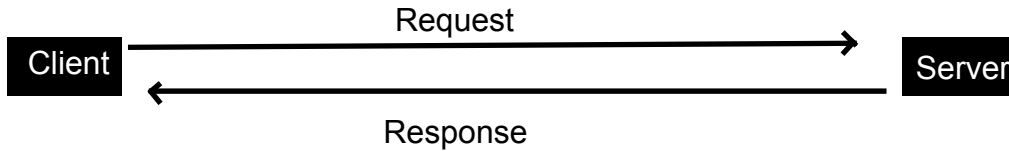
- **Standart Web Transfer Protocol:** Web üzerindeki herhangi bir dosyaya erişmek için kullanılan bir protokol

-**Http Port Numarası 80:** Http ile ilgili tüm trafiğin 80 numaralı bağlantı noktasına iletiliğın ve alındıgın anlamına gelir .

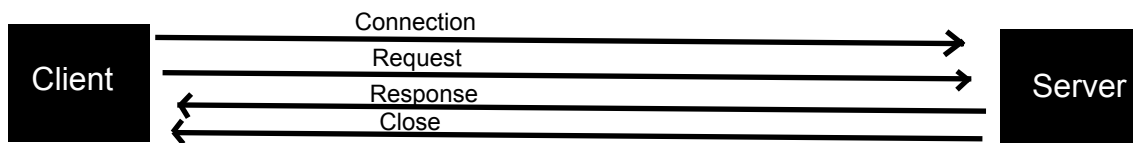
-**Transfer Hypermedia Document:** HTTP protokolü üzerinden metin,resim,ses,video ve diğır multimedya dosyaları gibi dosyaların web üzerinden aktarılmasını sağlar.

HTTP Nasıl Çalışır:

Http'nin çalışma kuralı, request-response davranışlarına dayalıdır. request-response davranışını anlamak için aşağıdaki görsele bakabiliriz



genelde yukarıda gördüğünüz görsel, internette herkesin çok sıklıkla gördüğü görsel çeşiti, fakat ben biraz daha olaya vakıf olabilme açısından, aşağıdaki görsel üzerinden gideceğim.



Connection: Bu başlık altında olanları anlamak için temel network bilmemiz gerekiyor çünkü burada gerçekleşen durumlar DNS, Three way handshake gibi durumlar, bunu burada anlatmayacağım fakat web sitemde bulabilirsiniz.

Request: Client tarafından sunucu tarafına bilgi isteği bulunma durumudur.

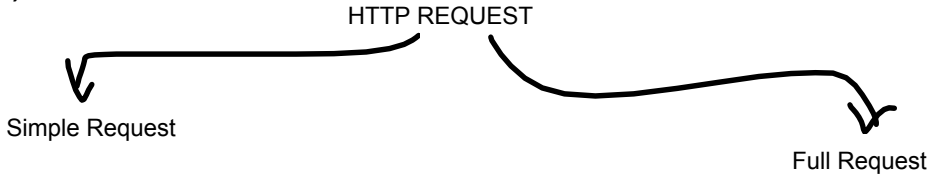
Response: Server tarafından,client tarafına bilgiyi döndürme durumudur.

Close: taraflardan biri veya her ikisi de işlemi sonlandırma durumudur.

Ben burada Request ve Response başlıklarını anlatacağım ama kesinlikle Connection kısmıda bilinmeli.

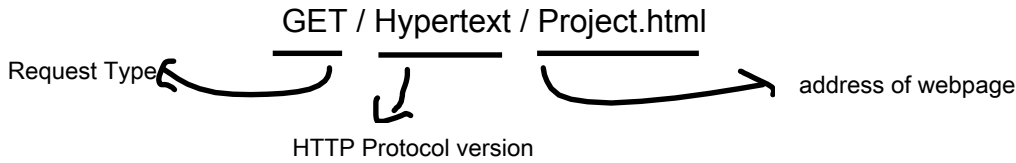
HTTP REQUEST:

Client tarafından, sunucu tarafına bilgi isteği bulunma durumu olduğunu söyledim, peki ama bu isteği nasıl belirliyor, öncelikle bunu anlatmadan önce HTTP request türlerin kendi içerisinde ikiye ayrıldığını söylemeyelim bunlar



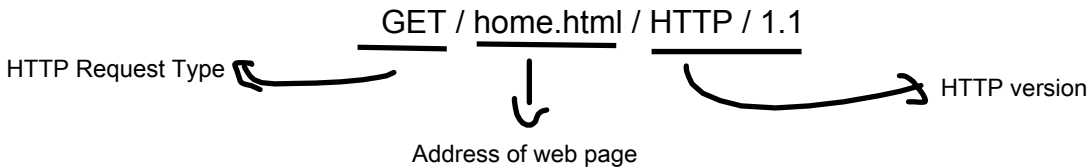
Simple Request:

Simple Request, protokol sürümü olmadan, istenen sayfayı adlandıran tek bir GET satırından ibarettir diyebiliriz. Bunu daha anlamlı kılmak için aşağıdaki Simple Request şeması bakabiliriz



Full Request:

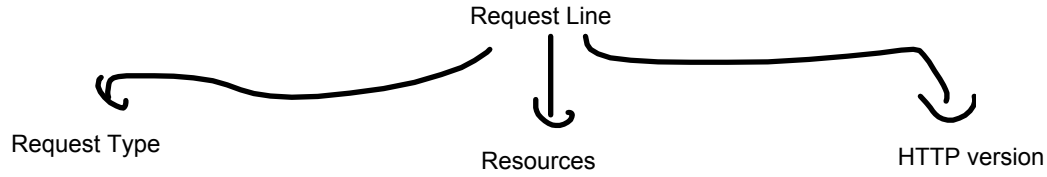
Full Request, GET istek satırında, protokol sürümünde gösterilir. Bunu daha anlamlı kılmak için aşağıdaki Full Request şemasına bakabiliriz.



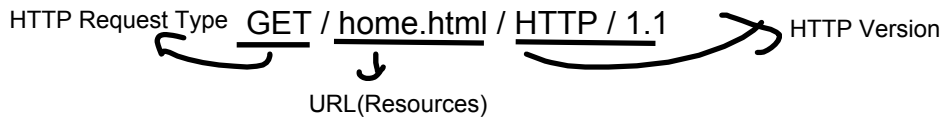
ve bundan yola çıkarak, http request hangi duruma göre oluyor sorusunun cevabını bilmek için, aşağıdaki http request mesaj formatının diyagramına bakmamız gerekli.

Request Line
Header Fields
Blank Line
Message Body

diyagramdaki başlıkların ne işe yaradığını tek tek bakacak olursak ve bunları da alt başlıklar haline özetlemek gerekirse



Ve bunları da anlayacağımız dile çevirecek olursak



Http Request Type: Client olarak, karşı tarafa yapacağımız istek türünü, Http istek tipleri ile belirtiriz, En çok kullanılan HTTP İstek tipleri, GET(almak için) ve POST(vermek). Diğer Http istek tiplerine bakmak isterseniz, <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods> bu adresinden bakabilirsiniz.

Resources: Url, Web dünyasında ilgili web sitesine erişmek için kullandığımız adresi tanımlar. Örnek üzerinden gidecek olursak.

http://www.facebook.com/home.html	
http	Protocol
www	Subdomain
facebook	Domain name
com	Top Level Domain
home.html	File Path

Tabi ki de böyle bir domain adresi bilgisayar okuyamadığı için, dns tarafından ilgili sunucunun IP adresi belirlenip, Http protokolu tarafından o adrese istek atılıyor. Ve böyle bir durumda aşağıdaki şema karşımıza geliyor.

<http://facebook.com> -----> <http://157.240.196.35/index.html>

burada ufak bir bilgi vermek istiyorum, url çubuğunda port numarasını görmemeizin nedeni, TCP protokolünde default olarak, 80 numaralı port numarasını belirtilmesi. Yukarıdaki http şemasını biraz daha açıklayıcı hale getirmek istersem.

Protocol://Host:Port/Path

HTTP Version: Kullanılan HTTP Sürümünü ifade eder. HTTP sürümlerinde, sürümler arasında farklılıklar var,bunu web sitemde bulabilirsin.

Diyagram üzerinden anlatılan başlıkları daha vakıf olabilmek açısından, aşağıda anlatılan diyagramın, gerçek dünya örneği mevcut

Request Line

```
GET /index.html HTTP/1.1\r\n
Host: [REDACTED]\r\n
User-Agent: Firefox/3.6.10\r\n
Accept: text/html, application/xhtml+xml\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip,deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7\r\n
Keep-Alive: 115\r\n
Connection: keep-alive\r\n
\r\n
```

Blank Line

HTTP RESPONSE:

Sunucu tarafından, Client tarafına bilgiyi döndürme durumunu olduğunu söylemiştik, Http Response başlıklarının, Http Request başlıklarına göre biraz farklılıkları mevcut.

Status Line
Header Fields
Blank Line
Message Body

Status Line: Sunucu tarafından, Client tarafına gönderilen yanıt başlığı. Client tarafı isteğin başarılı olup olmadığını bu status kodları ile anlıyor. Genellikle durum kodları .

1xx status = Bilgi için döndürülen durum kodları.

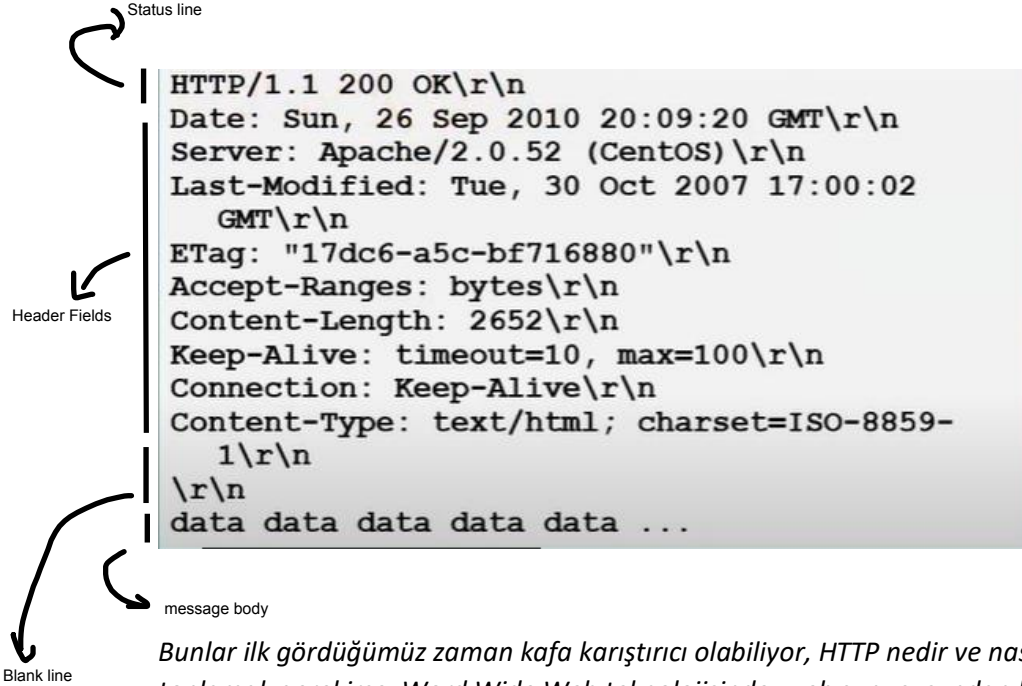
2xx status = isteğin başarılı olduğunu ifade eden durum kodları.

3xx status = yönlendirme(Redirect) içeren durum kodları.

4xx status = client tarafında hata olduğunu bildiren durum kodları.

5xx status = sunucu tarafında hata olduğunu bildiren durum kodları.

Bu durum kodlarına ayrıntılı bir şekilde öğrenmek istiyorsanız, <https://developer.mozilla.org/enUS/docs/Web/HTTP/Status> adresin üzerinden bakabilirsiniz. Ve aşağıda ise, yukarıdaki diyagramın, şema ve gerçek dünya üzerindeki bir http response ağacına bakarak olaylara daha vakıf olacağınıza inanıyorum.



Bunlar ilk gördüğümüz zaman kafa karıştırıcı olabiliyor, HTTP nedir ve nasıl çalışır sorusunun cevabını toplamak gerekirse, Word Wide Web teknolojisinde, web sunucusundan bilgi almak istediğimiz zaman kullandığımız protokol ve bu protokol Http İstek Türleri ve Http Durum kodları aracılığıyla birbirleri ile iletişim kurar diyebiliriz.

e-mail

berathanakcakaya@gmail.com

linkedin

<https://www.linkedin.com/in/berathan-akcakaya/>