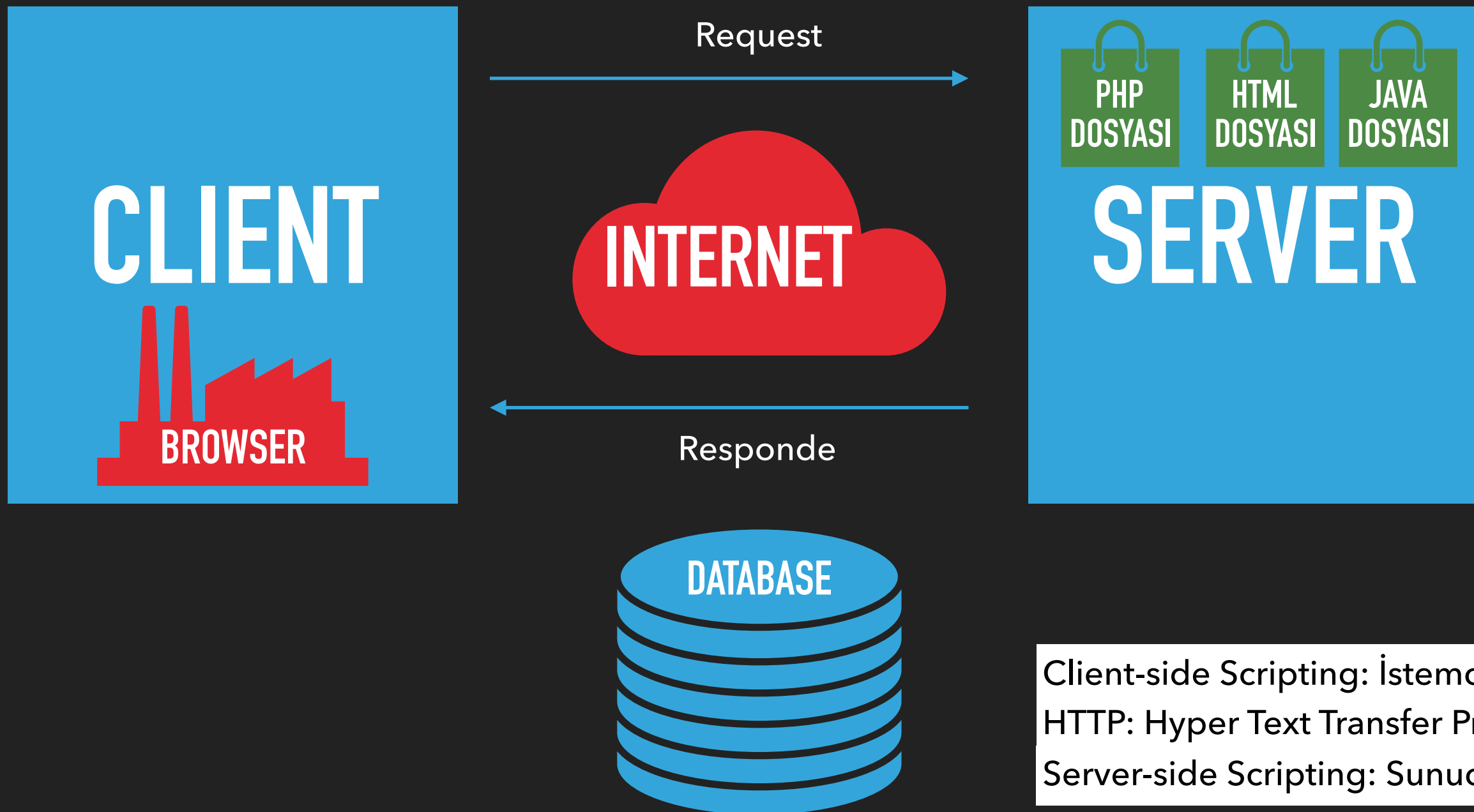


HTML & CSS & JAVASCRIPT

WEB & MOBİL EĞİTİMLERİ

WEB NASIL ÇALIŞIR?



Client-side Scripting: İstemci betik dili
HTTP: Hyper Text Transfer Protocol
Server-side Scripting: Sunucu betik dili

HTML NEDİR?

- ▶ HTML (Hypertext Markup Language)
- ▶ Bir programlama dili değil
- ▶ Tarayıcı ile iletişim
- ▶ Web sayfalarının iskeleti
- ▶ Etiketler `<>` `</>`

CSS NEDİR?

- ▶ CSS temelde HTML elemanlarının stillendirir
- ▶ Bir programlama dili değil
- ▶ Sayfanın görünümünü değiştiririz
- ▶ Etiketler kullanır `<>` `</>`

JAVASCRIPT NEDİR?

- ▶ Scripting dili olarak tasarlanmıştır
- ▶ Yorumlanan bir dildir ve yorumlayıcının olduğu her ortamda çalışır (NodeJs, Electron)
- ▶ Çalışmak için Javascript Motorlarından birine ihtiyaç duyar
- ▶ 1995 yılında 10 günde yazılmıştır :)
- ▶ Chrome – V8 | Firefox – SpiderMonkey | Safari – JavaScriptCore gibi motorlar kullanır
- ▶ EcmaScript dili standartının bir uygulamasıdır

The World Wide Web project

WORLD WIDE WEB

The WorldWideWeb (W3) is a wide-area hypermedia[1] information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary[2] of the project, Mailing lists[3] , Policy[4] , November's W3 news[5] , Frequently Asked Questions[6] .

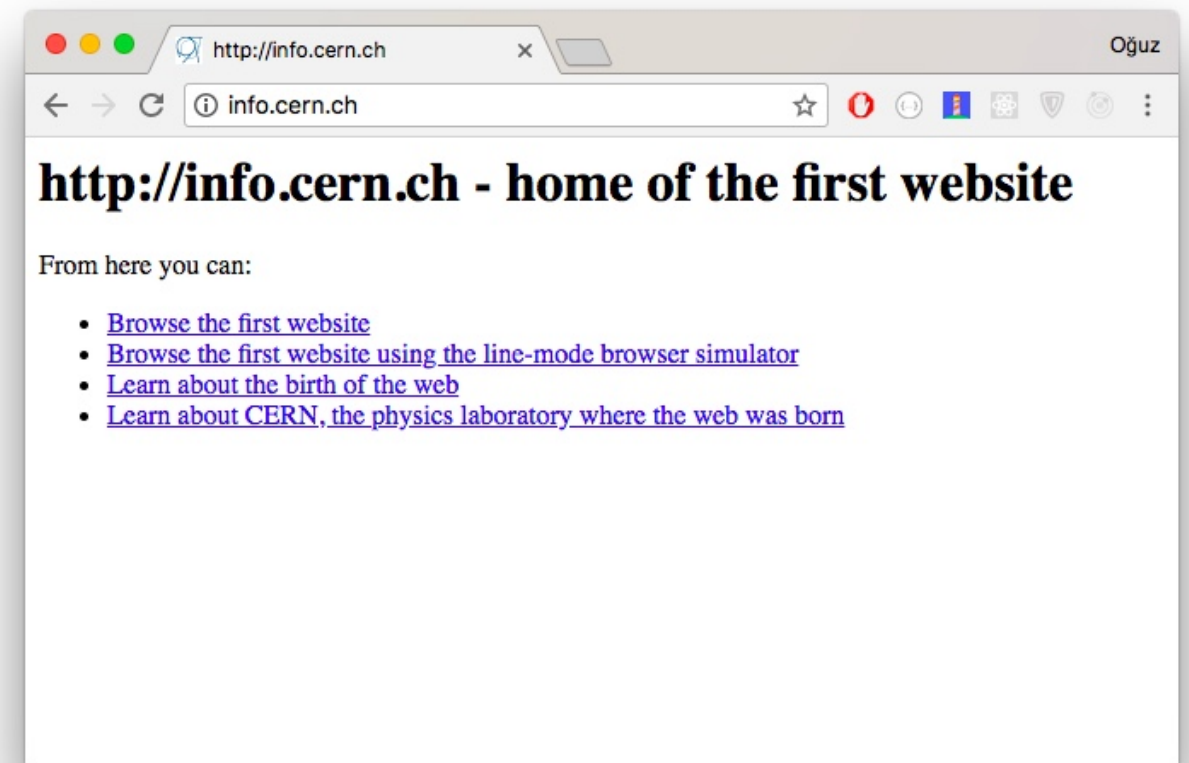
What's out there?[7]Pointers to the world's online information, subjects[8] , W3 servers[9], etc.

Help[10] on the browser you are using

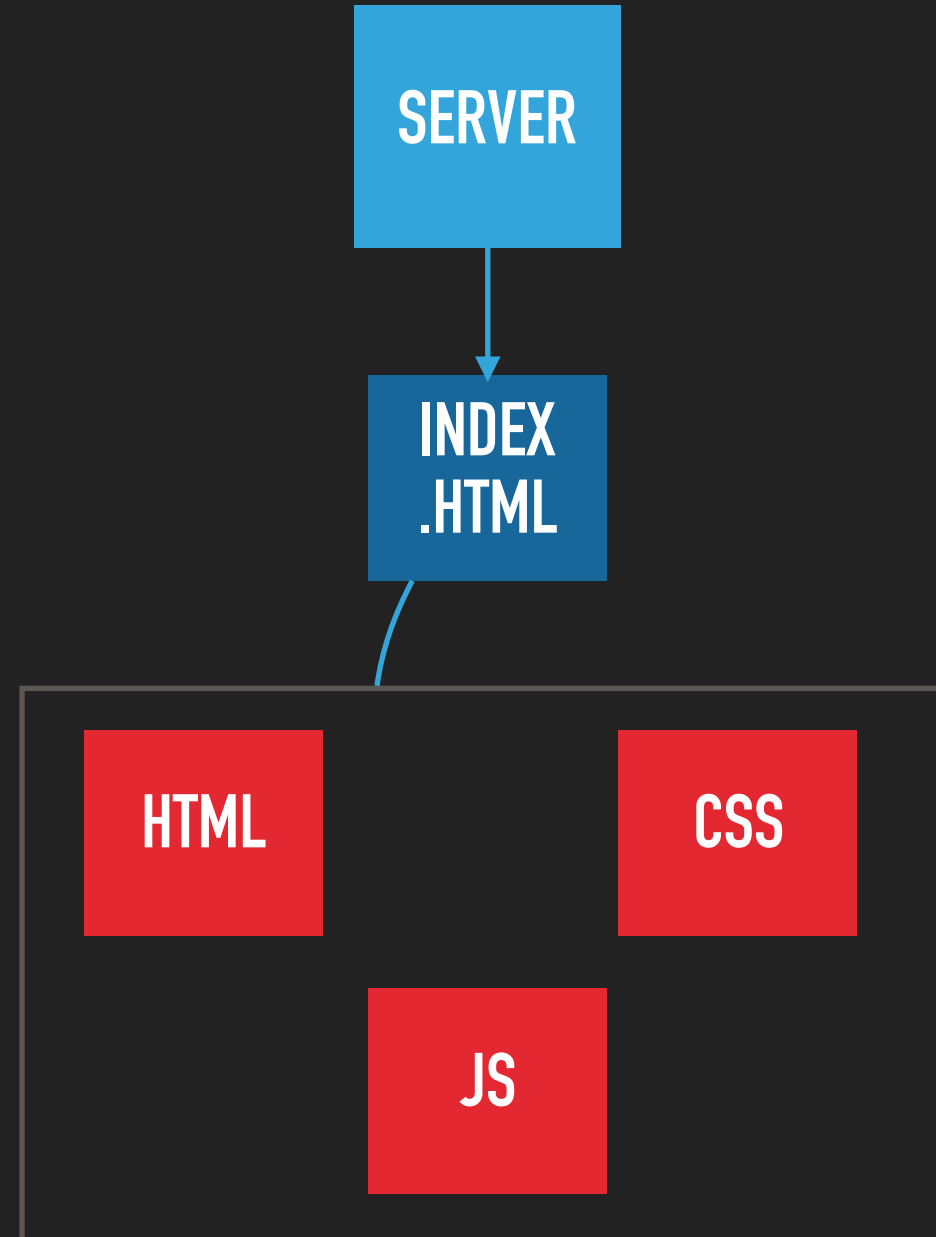
Software Products[11] A list of W3 project components and their current state. (e.g. Line Mode[12] ,X11 Viola[13] , NeXTStep[14] , Servers[15] , Tools[16] , Mail robot[17] , Library[18])

Technical[19] Details of protocols, formats, program internals etc

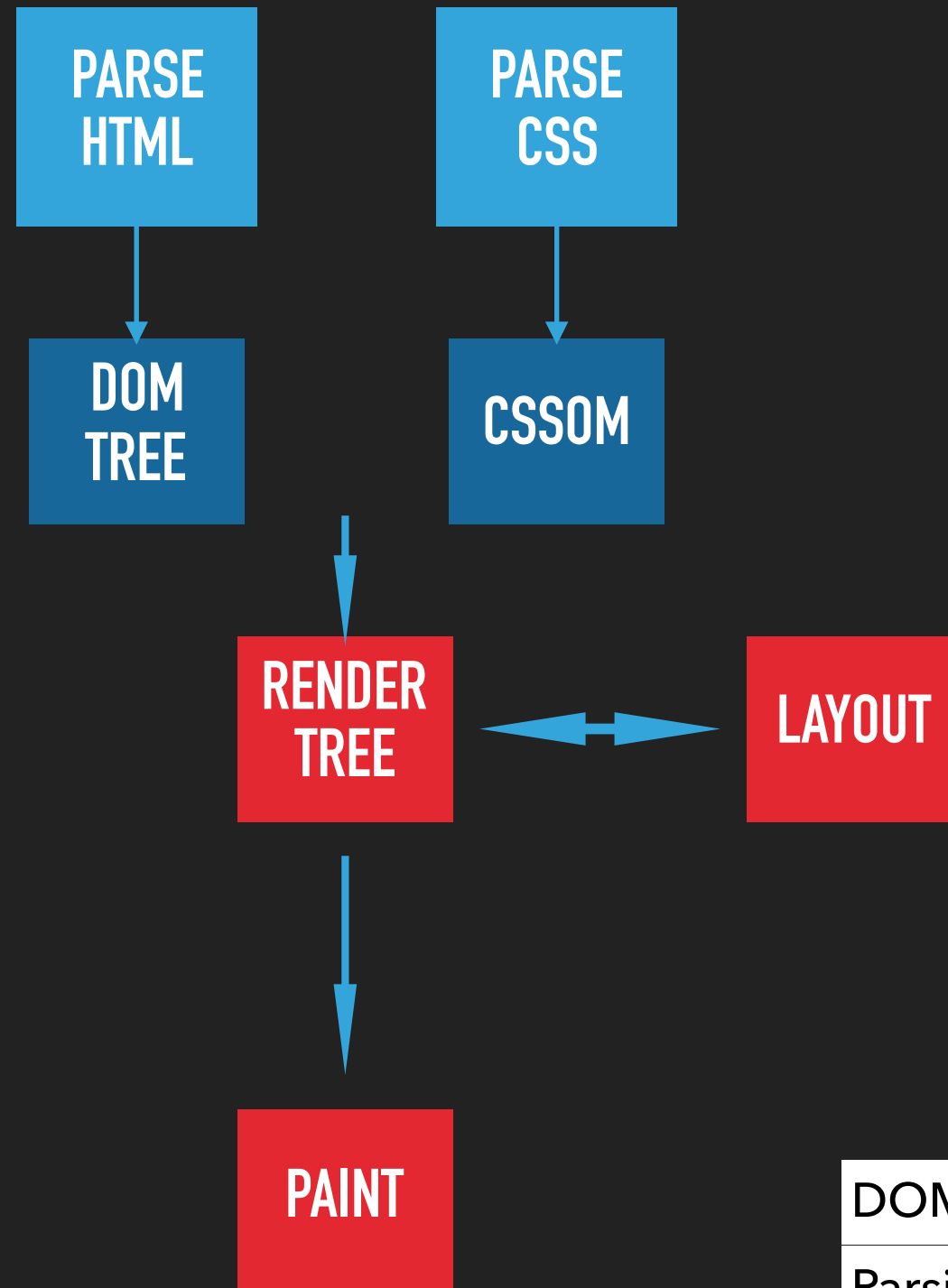
<ref.number>, Back, <RETURN> for more, or Help: █



INDEX.HTML İÇERİĞİ



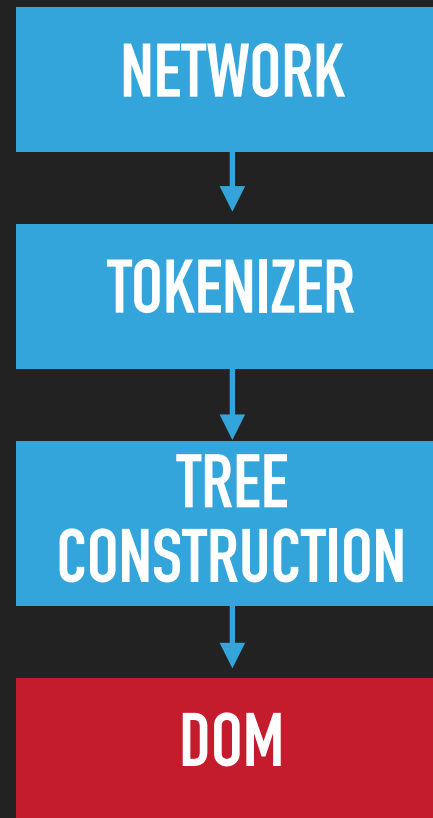
INDEX.HTML NASIL YORUMLANIR?



DOM: Document object model

Parsing: Çözümlemek, incelemek

HTML PARSE EDİLME İŞLEMİ?

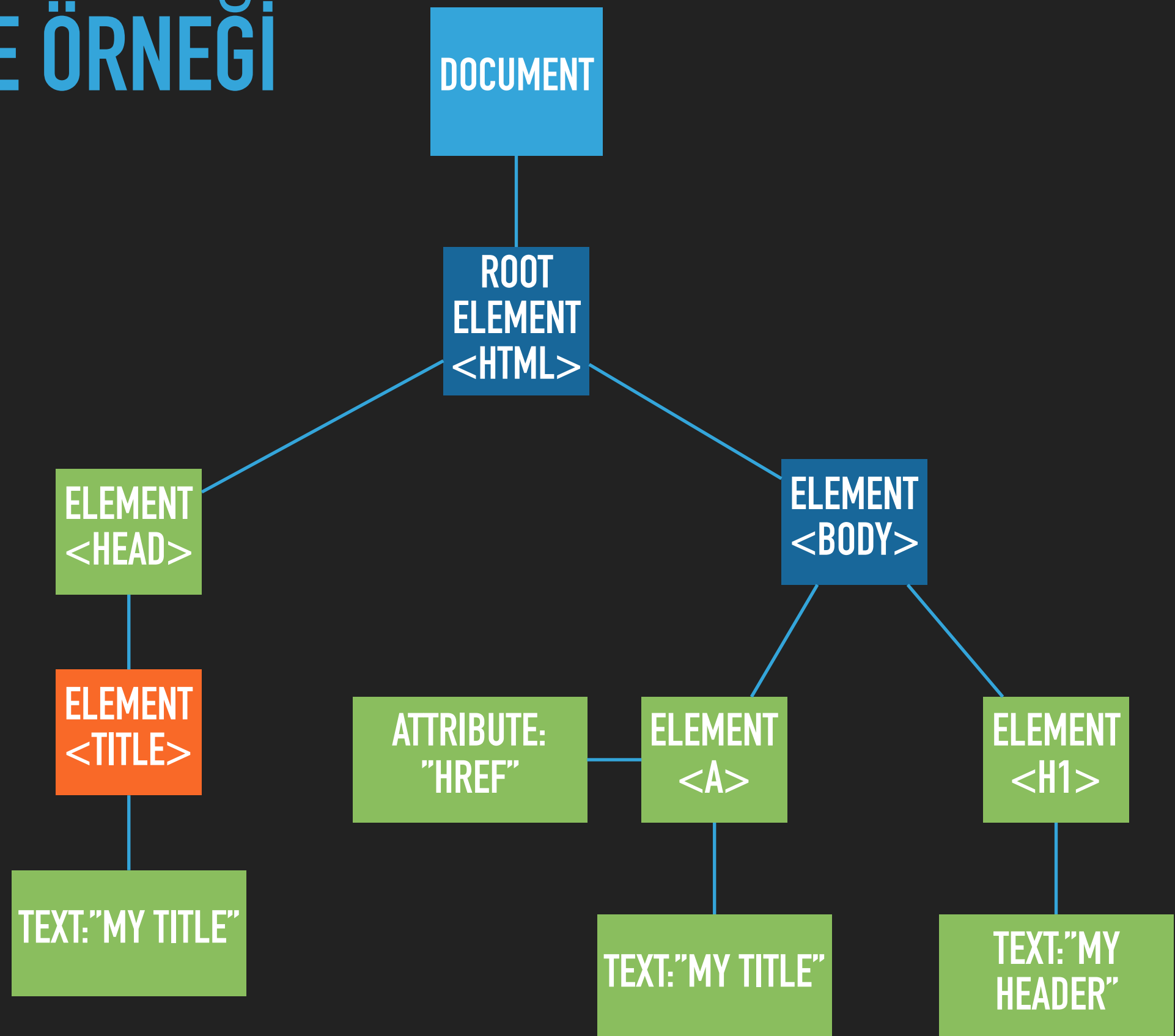


DTD: Document Type Definition

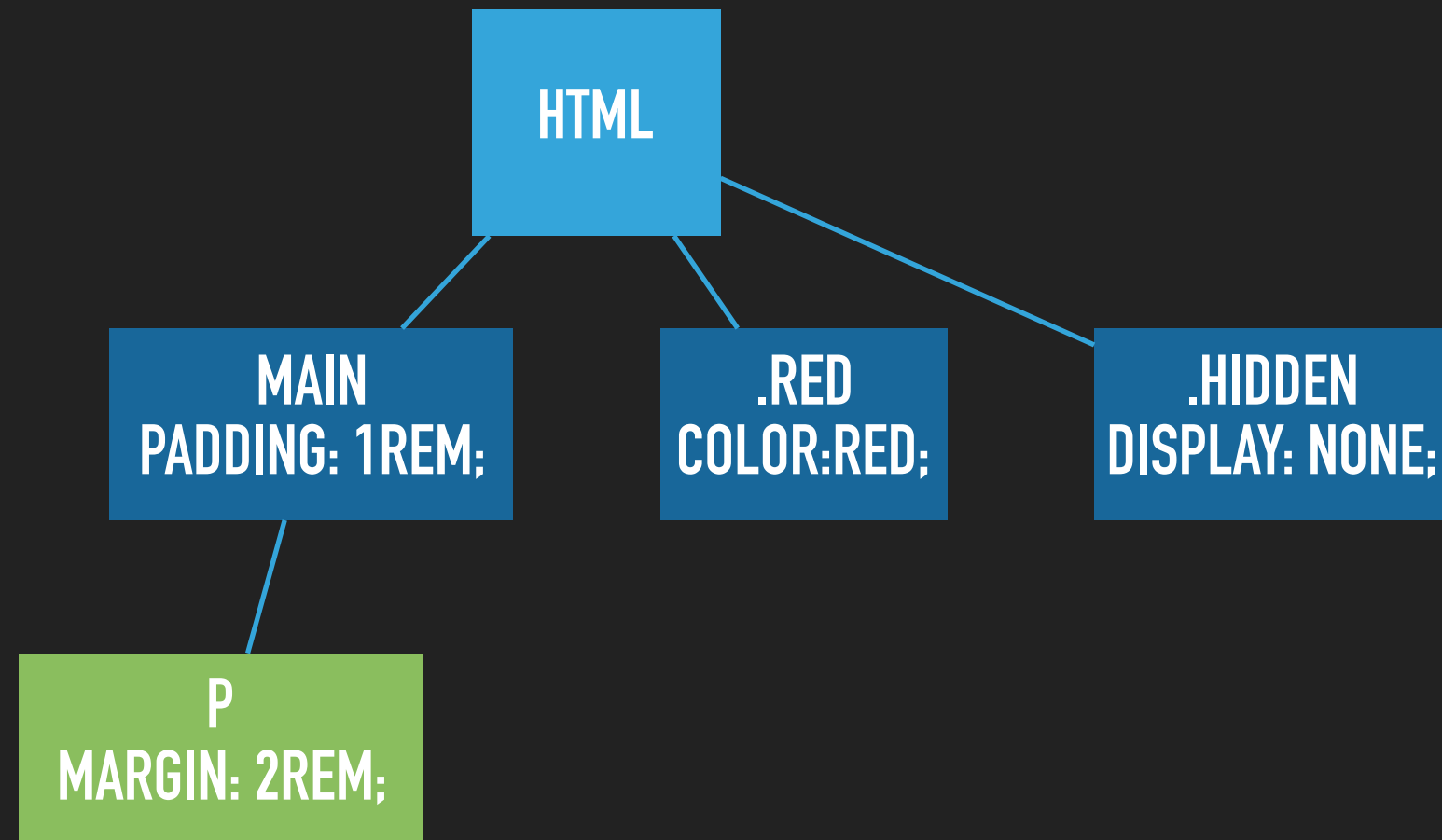
Chunk: İri veri parçaları

Tokenizing: Atomik yapılara ve simgelere ayırm

DOM TREE ÖRNEĞİ

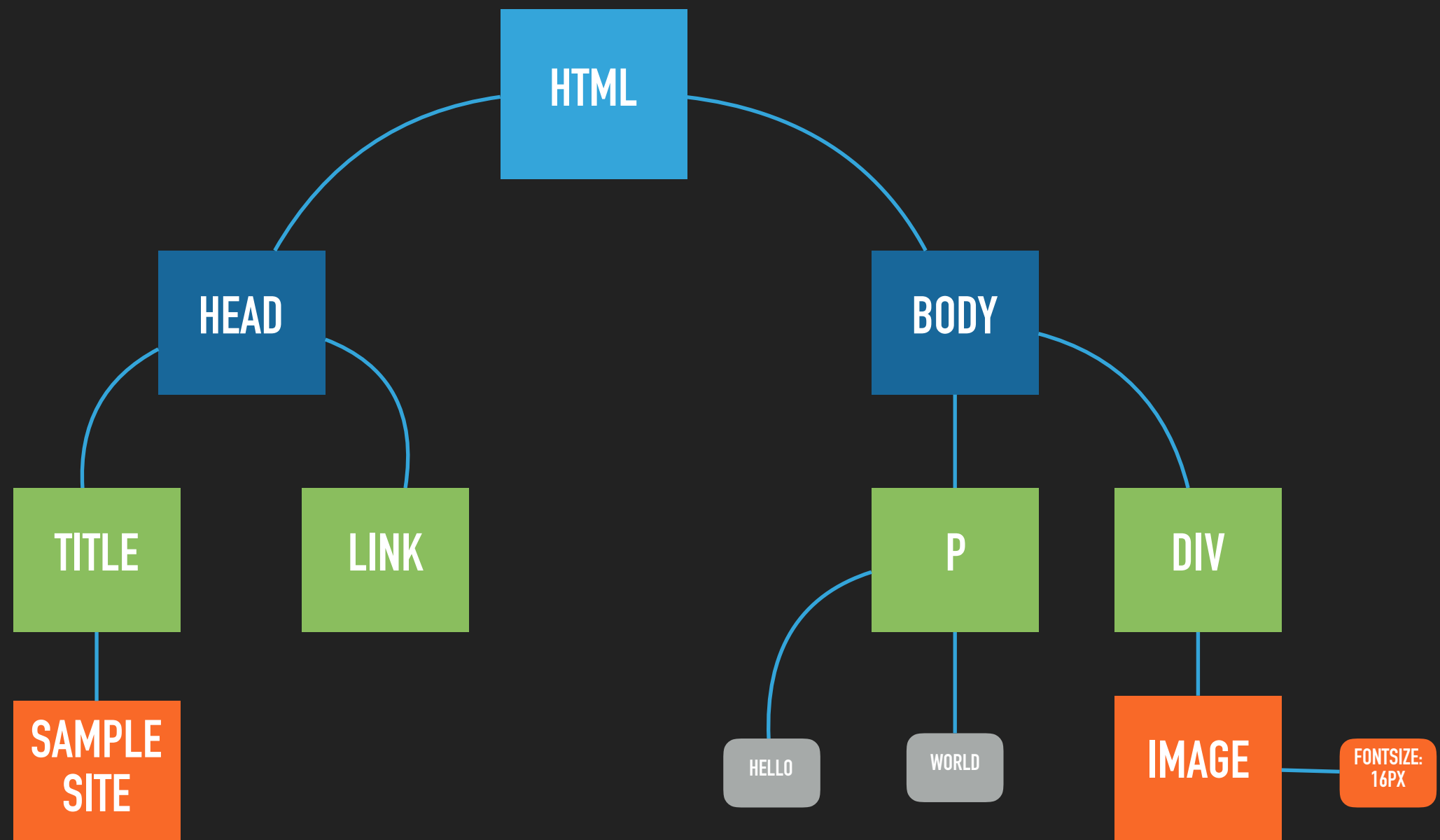


CSSOM ÖRNEĞİ



RENDER TREE NEDİR?

Render Tree Örneği



HTML

HTML KOD YAPISI

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sitenin Temel Özellikleri</title>
  </head>
  <body>
    <p>Sitenin İçerik Kısımı</p>
  </body>
</html>
```

ÖNEMLİ ETİKETLER

- ▶ Başlıklar (Heading) -> `<h1></h1>` | `<h2></h2>` | `<h3></h3>`
- ▶ Paragraflar (Paragraphs) -> `<p></p>`
- ▶ Bağlantılar (Links) -> `Mesaj.`
- ▶ Görseller (Images) -> ``
- ▶ Div (Division) -> `<div> </div>`
- ▶ Sayfanın hangi kurallara göre parse edildiği? -> `<!DOCTYPE html>`
- ▶ Tablolar (Tables) -> `<th>(Header)` | `<tr>(Row)` | `<td>(Cell)` | `<caption>(Title)`
- ▶ Tanımlama etiketi -> `id="demo"`

PARAGRAF ETIKETLERİ

- ▶ Kalın Yazı -> ``
- ▶ Yatık Yazı -> ``
- ▶ Üstü Altı Çizilmiş Yazı -> `<strike></strike>` | ``
- ▶ Büyük Küçük Yazı -> `<small> </small>` | `<big></big>`
- ▶ Satır Atlamak ve Yatay Çizgi -> `
 </br>` | `<hr> </hr>`
- ▶ Liste Etiketi -> `` | ``

CONTAINER NEDİR?

- ▶ Web sitesini düzenli tutmak için kullanılan bir yapı



CONTAINER NEDİR?

- ▶ Header -> Site logosu ve dahili öğeler
- ▶ Nav -> Sitenin arama öğelerini barındıran kısım
- ▶ Article + Section -> Sitenin içerik kısmı
- ▶ Aside -> Sitenin ekstra bilgiler kısmı
- ▶ Footer -> Web sitenin yapılış, iletişim ve telif bilgileri

css

CSS KOD YAPISI

```
P{  
  color: red;  
  text-align: center;  
}
```

CSS Kodları `<style></style>` etiketi içinde yazılır

ÖNEMLİ ETIKETLER

- ▶ Renk (Color) -> `color: #999 | background-color: green | opacity: 0.5`
- ▶ Font (Font) -> `font-size: 120 | text-transform: uppercase | text-transform: lowercase`
- ▶ Kenarlar (Margins) -> `margin-left: 50% margin-right: 50% margin-top: 50% | border: 5px solid red`
- ▶ Dolgu (Padding) -> `auto |padding-top: 50px | padding-right: 30px; | padding-left: 80px;`
- ▶ Görünebilirlik & Gösterge (Visibility & Display) -> `Display: None & Visibility: Hidden`

JS

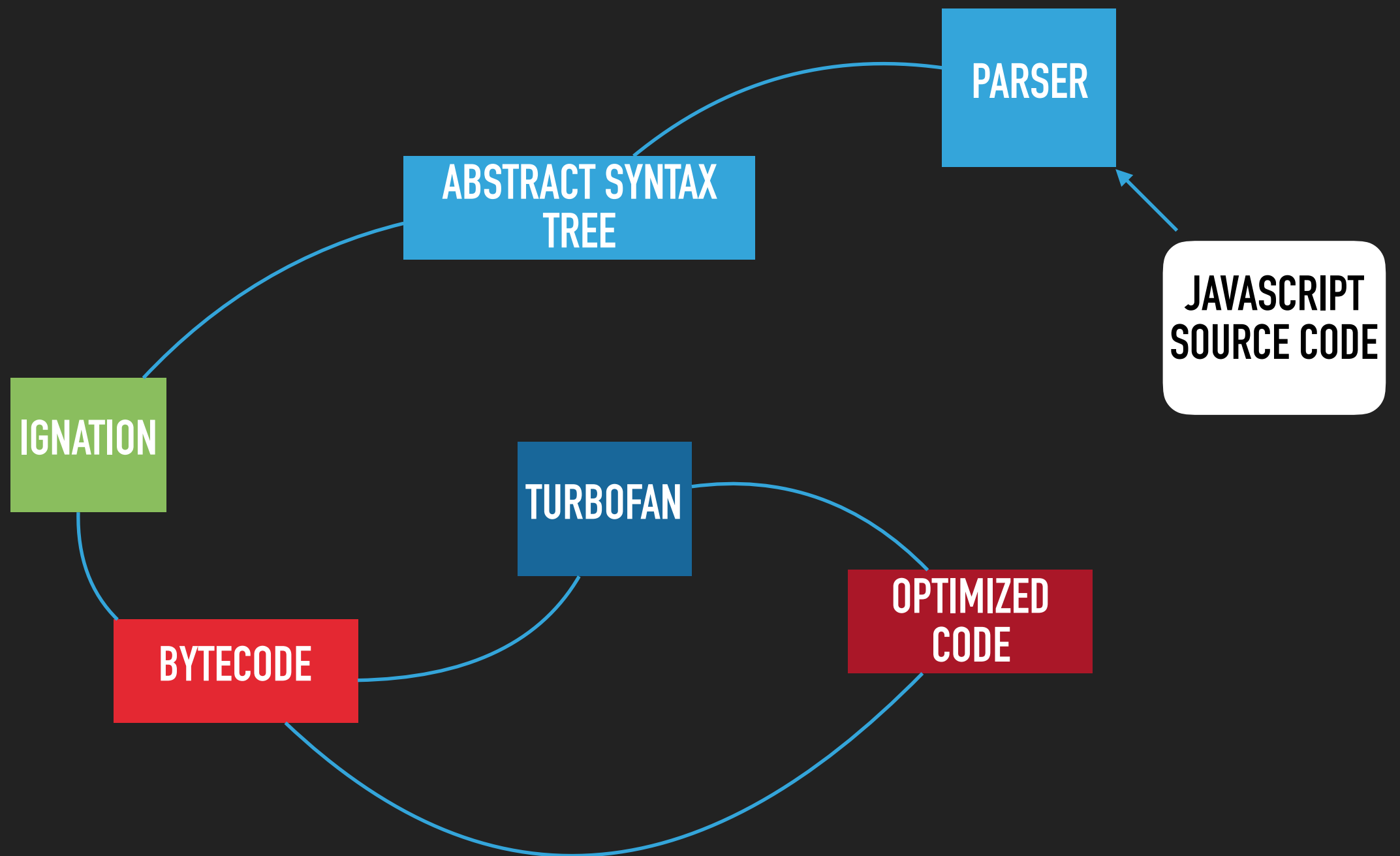
ÖNEMLİ ETİKETLER

- ▶ Belirli bir ideye ulaşmak için -> `document.getElementById()`
- ▶ Typecasting -> `parseInt()` | `parseFloat()`
- ▶ Yorum (Comment) -> `//`
- ▶ Dolgu (Padding) -> `padding-top: 50px` | `padding-right: 30px;` | `padding-left: 80px;`
- ▶ Görünebilirlik & Gösterge (Visibility & Display) -> `Display: None` & `Visibility: Hidden`
- ▶ Sayfanın hangi kurallara göre parse edildiği? -> `<!DOCTYPE html>`
- ▶ Kısım etiketi -> `<section></section>`

JAVASCRIPT MOTORLARI?

- ▶ Javascript kodunu yorumlamak ve çalıştırmak için özel tasarlanmış bir PVM'dirler
- ▶ Javascript motorları basitçe **analiz**, **yorumlama**, **optimizasyon** ve **garbage collector** gibi yapılara sahiptir
- ▶ EcmaScript standartlarını motorlar kendileri implemente ederler. Bu noktada geriye uyumluluk için kodumuz transpile denilen araçlardan geçirilir. En yaygını **babel**

JAVASCRIPT V8 ENGINE?



TurboFan: JIT Derleyici
Ignation: Yorumlayıcı

AST NEDİR?

- ▶ Javascript kodunun analiz çıktılarıdır
- ▶ Sonrasında bu çıktılar makine koduna dönüştürülür

Örneğin

```
const ast = "Oğuz";
```

```
✓ Autofocus ✓ Hide methods ✓ Hide empty keys ✓ Hide location data ✓ Hide type keys
- Program {
  sourceType: "module"
  - body: [
    - VariableDeclaration {
      - declarations: [
        - VariableDeclarator {
          + id: Identifier {name}
          - init: Literal {
            value: "Oğuz"
            rawValue: "Oğuz"
            raw: "\"Oğuz\""
          }
        }
      ]
    }
  ]
  kind: "const"
}
+ tokens: [6 elements]
comments: [ ]
}
```

TREE SHAKING

Yazdığınız kodların içerisinde

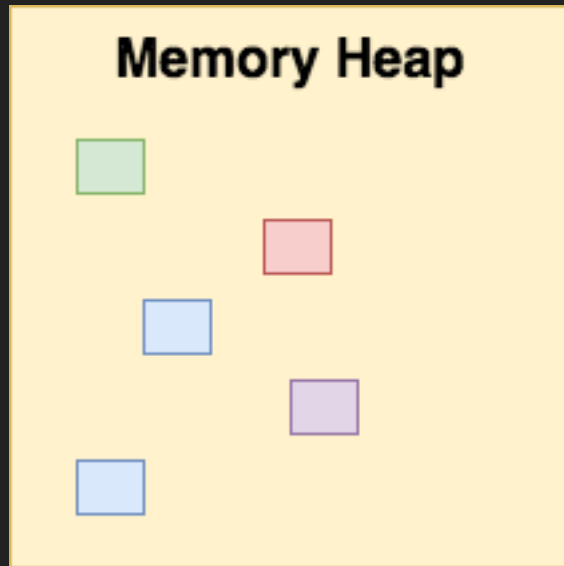
Kullanılmayan kodların elimine edilmesi

için kullanılan bir terim. Performans

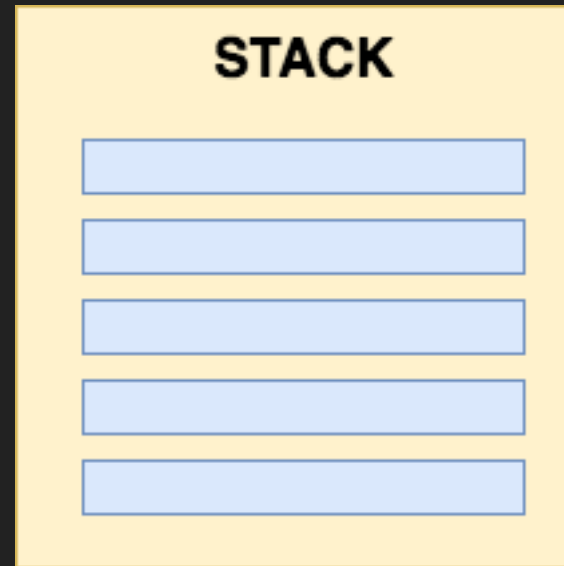
avantajı sağlıyor

AST: Abstract Syntax Tree

JAVASCRIPT EŞZAMANLILIĞI?

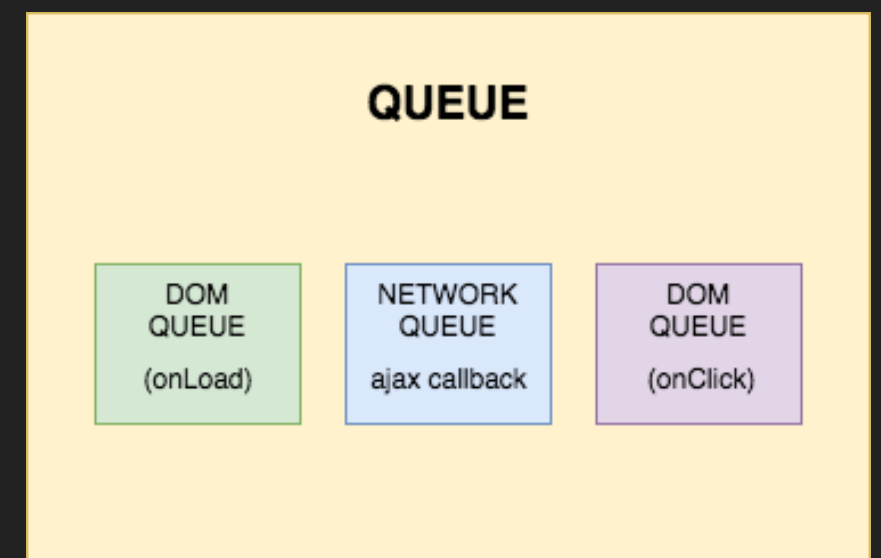


Bellek ayırma süreçlerinin
gerçekleştiği yer



Anlık olarak nerede olduğumuzu
Kaydeden bir veri yapısı denilebilir
Javascript single-thread bir dil olduğu
için tek bir call-stack yapısına sahip.

Yani aynı anda sadece bir iş yapılabilir



Javascript runtime'ında işlenecek
iletiler bir queue halinde tutulur

JAVASCRIPT GARBAGE COLLECTION YAPISI?

- ▶ Garbage Collector bellek optimizasyonu yapan bir yapıdır
- ▶ Periyodik olarak Javascript uygulaması için gereken belleğin üzerinden geçerek gerek olmayan yerleri temizler ve ciddi performans avantajları sağlar
- ▶ Başka bir yandan bakarsak bu işlemlerin otomatik yapılması ciddi kayıplara da sebep olabilir

JAVASCRIPT KOD YAPISI

- ▶ Javascript kodları <script></script> etiketi içinde yazılır.
- ▶ Değişken tanımlanabilir.
- ▶ Sayfayı yenilemeye gerek kalmaksızın değişiklik yapmak.

```
<script>  
  
var name = "Fatih";  
  
var a,b;  
  
a = 0;  
  
b = 0;  
  
</script>
```

SON