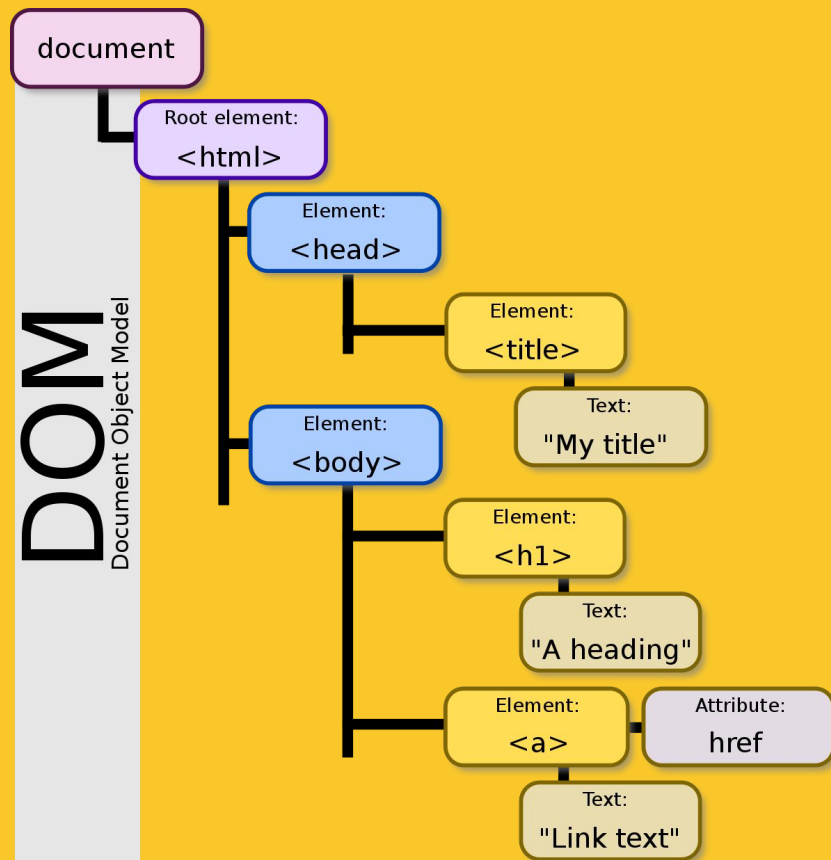


# DOM

## Document Object Model



## Apa?

DOM adalah **antarmuka / *interface* untuk HTML** yang merepresentasikan halaman Web sehingga HTML bisa dimanipulasi dengan Javascript

## Kenapa?

Untuk interaksi antara Javascript dan HTML

# Bagaimana?

Dengan cara menjadikan elemen HTML menjadi sebuah variabel / object di Javascript menggunakan DOM Selection

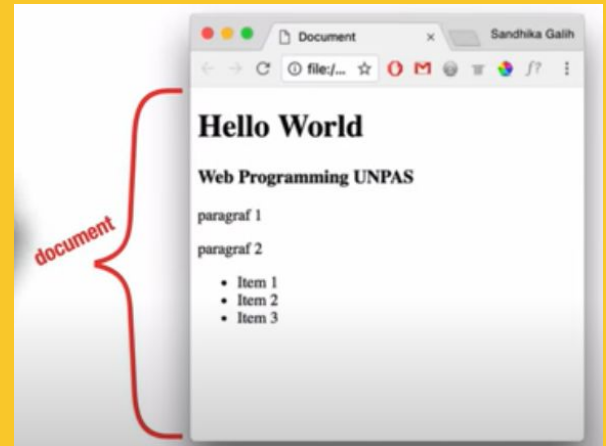
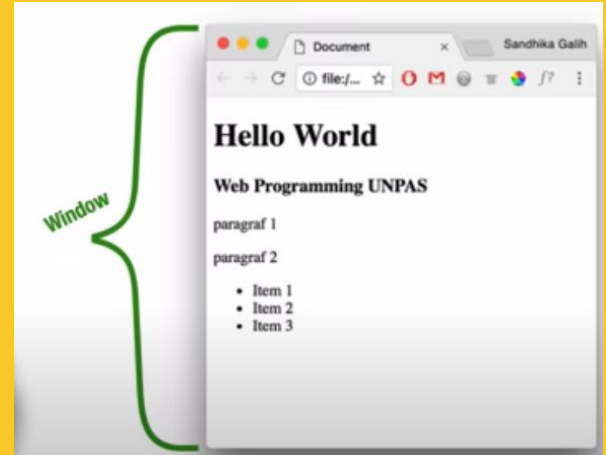
```
document.getElementById( )  
document.getElementsByTagName( )  
document.getElementsByClassName( )
```

```
document.querySelector( )  
document.querySelectorAll( )
```

# Apa itu document?

Adalah semua yang berinteraksi dengan kita dari HTML

Scope document di dalam area putih



# Apa itu Object?

Adalah salah satu tipe variabel dalam javascript

```
let object1 = {
```

```
  nama : gama,  
  github: @farsikogama,  
  ....
```

```
}
```

# Apa itu Model?

Representasi HTML di dalam Javascript

```
let div = {
```

```
  class : container,  
  id: #card,  
  href: 'www.ddd'  
  ....
```

```
}
```

# Menu Belajar DOM?

DOM Tree

DOM Selection

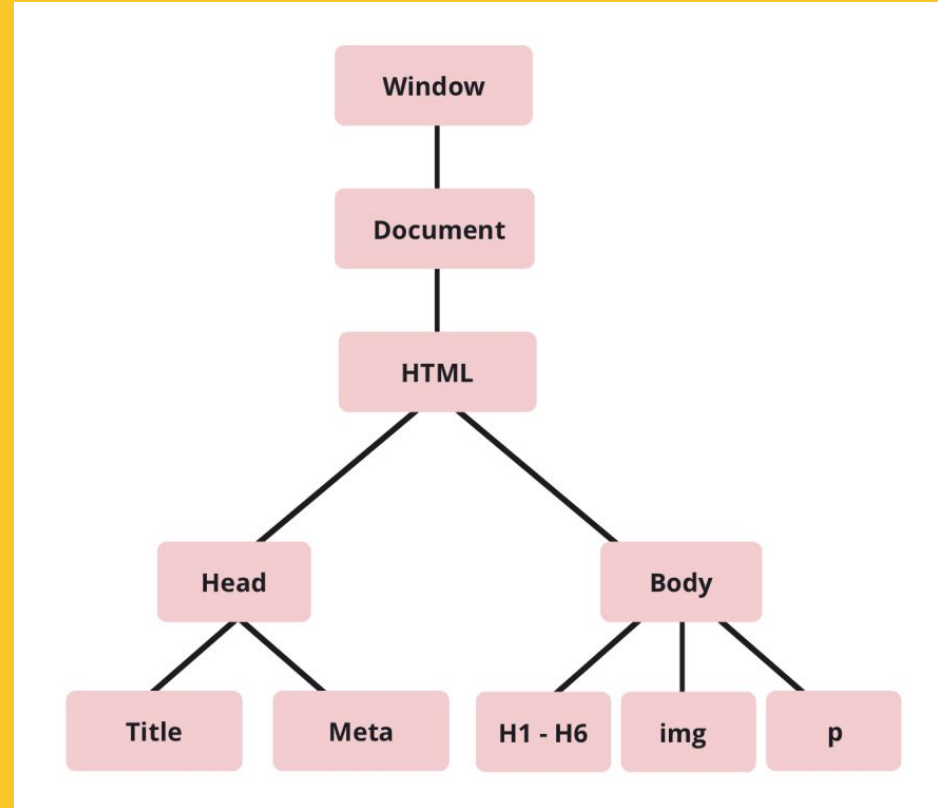
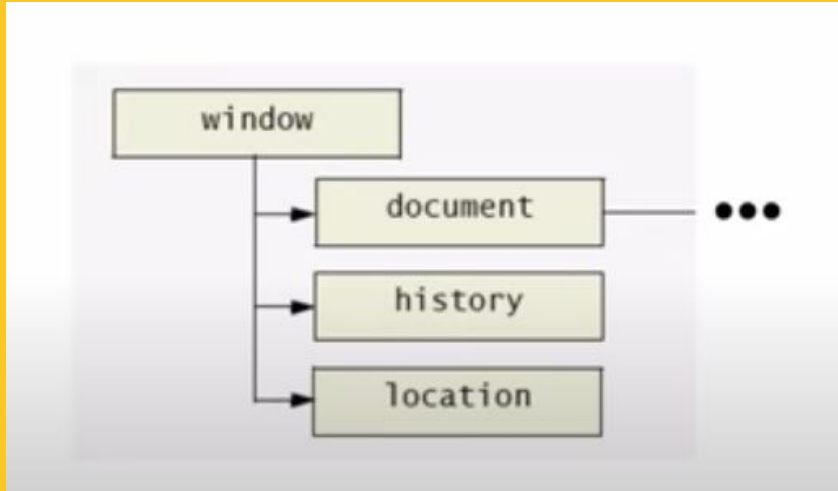
DOM Manipulation

DOM Traversal

Event Handling

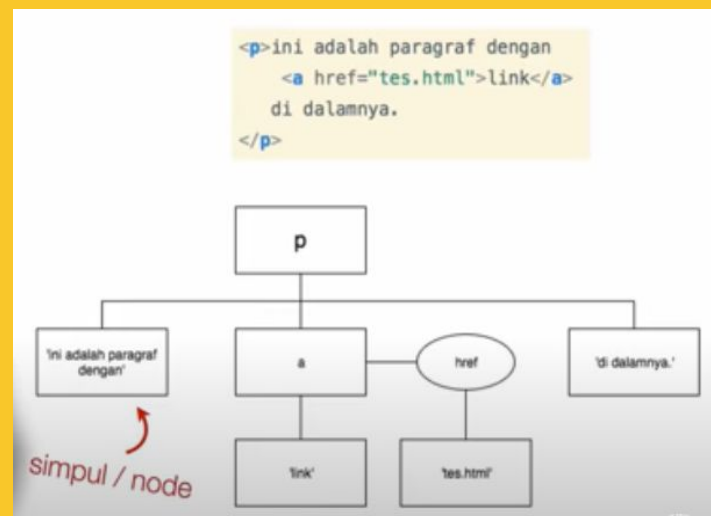
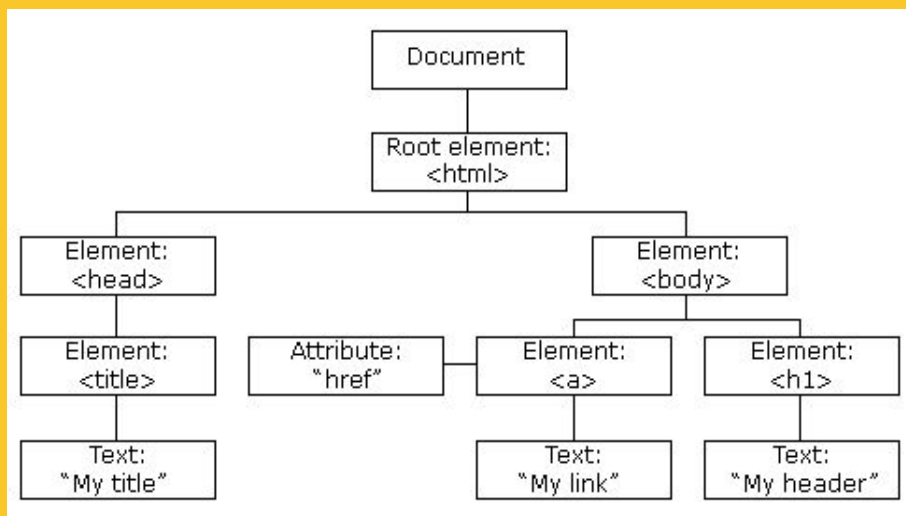
# DOM Tree

# Representasi HTML di dalam Javascript





# Tipe Node pada DOM Tree



# Type Node pada DOM Tree

Elemen  
Attribute  
Text

CData Section  
Entity Reference  
Entity  
Processing Instruction  
Comment  
Document  
Document Type  
Document Fragment  
Notation

**Node**

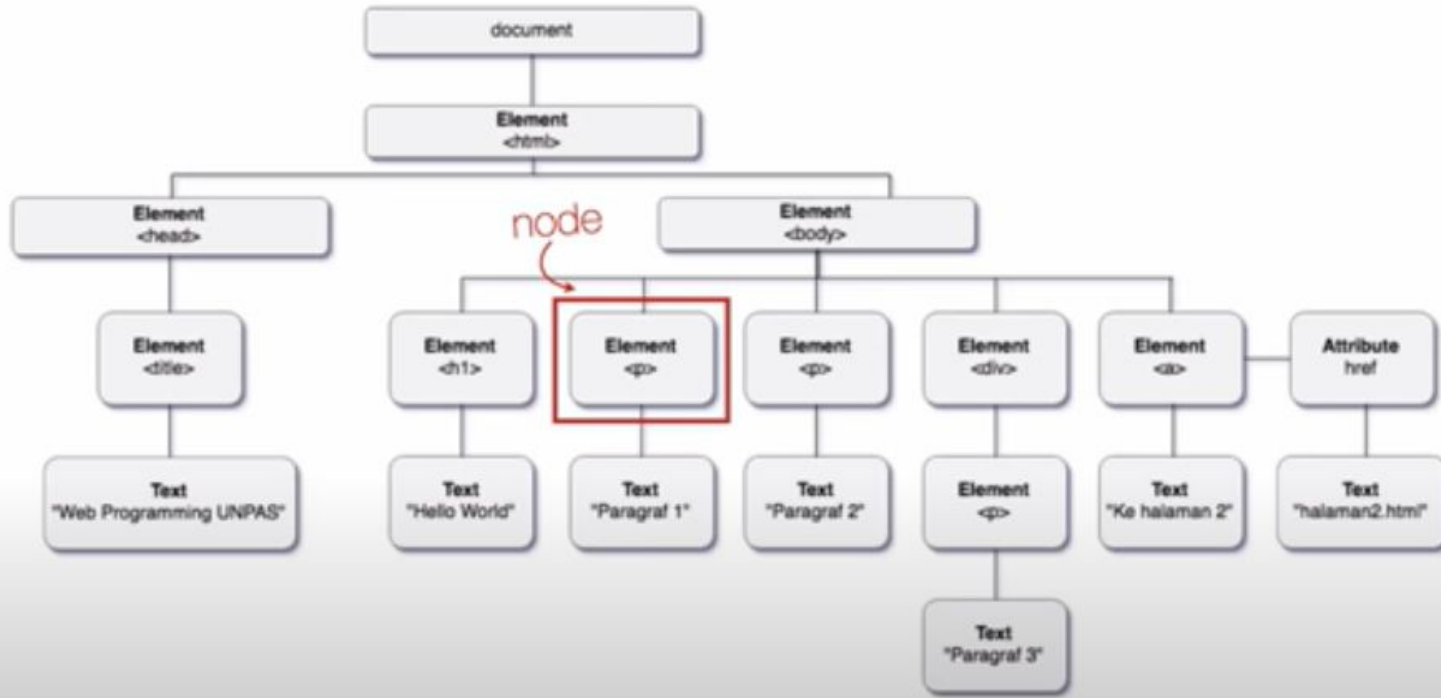
VS

**Node List**

VS

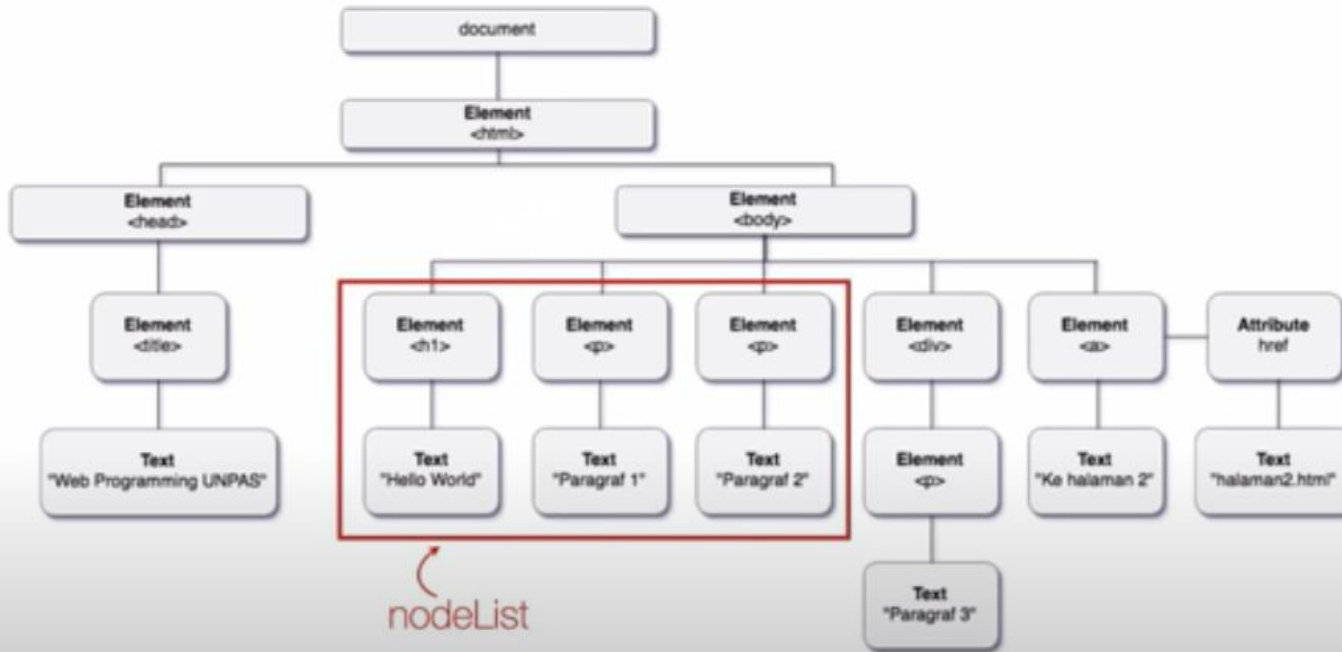
**HTML Collection**

# Node Adalah



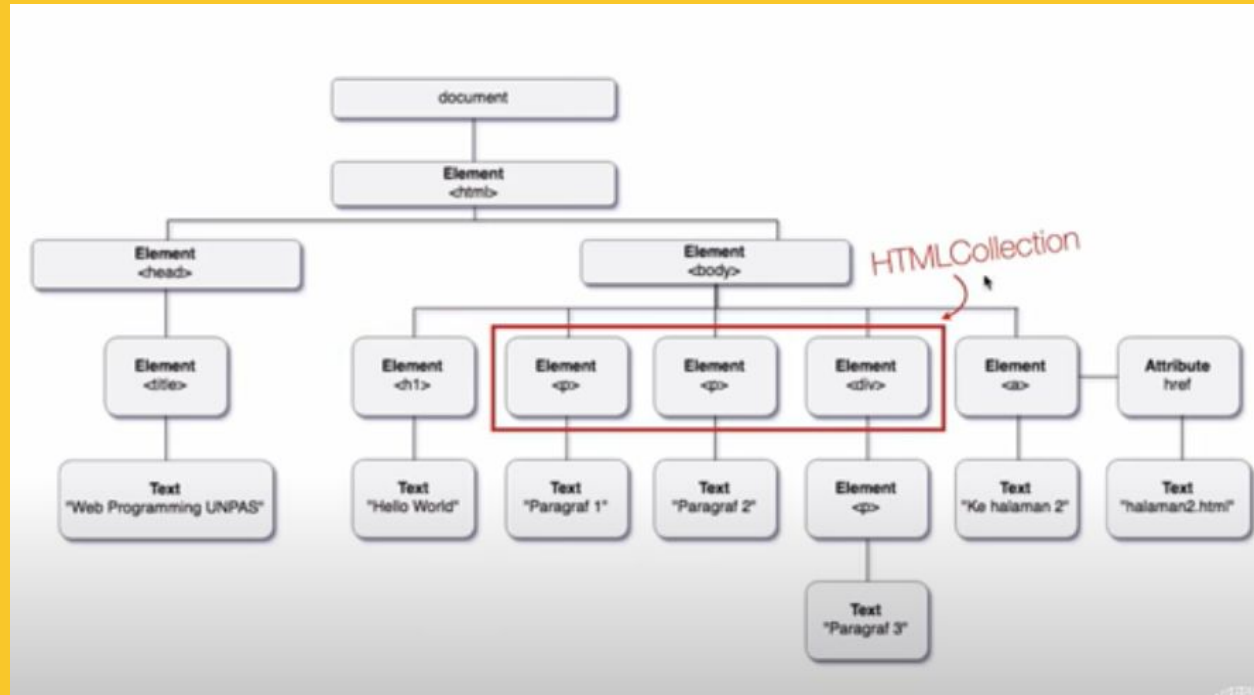
# Node List Adalah

Kumpulan Node tidak peduli tipenya apa



# HTML Collection Adalah

Kumpulan Node hanya yang bertipe Elemen



# Node List VS HTML Collection

## Persamaan

- Merupakan kumpulan node
- Struktur data mirip array

## Perbedaan

- Node List dapat berisi apa pun, sedangkan HTML Collection hanya berisi element
- Nodelist bisa dimanipulasi dengan beberapa method array
- HTMLCollection hanya dapat dimanipulasi dengan array method `array.map()`

# Struktur Hierarki DOM

## Root Node

- Node yang menjadi bungkus bagi semua node
- Document

## Parent Node

- Node yang berada satu tingkat di atas node lain
- Body adalah parent dari h1, p, div dll

## Child Node

- Node yang berada satu tingkat di bawah node lain
- H1 adalah child dari body



# DOM Selection

# DOM Selection Method

Method	Hasil
<code>getElementById()</code>	element
<code>getElementsByTagName()</code>	HTMLCollection
<code>getElementsByClassName()</code>	HTMLCollection
<code>querySelector()</code>	element
<code>querySelectorAll()</code>	nodeList

# Latihan

## Cek Vs Code

# **DOM**

# **Manipulation**

# Manipulasi Element

- `element.innerHTML`
- `element.style.<property>`
- `element.setAttribute()`
- `element.classList`
- ...

# Manipulasi Element

```
element.classList.add()
```

```
element.classList.remove()
```

```
element.classList.toggle()
```

```
element.classList.item()
```

```
element.classList.contains()
```

```
element.classList.replace()
```

# Manipulasi Node

- `document.createElement()`
- `document.createTextNode()`
- `node.appendChild()`
- `node.insertBefore()`
- `parentNode.removeChild()`
- `parentNode.replaceChild()`
- ...

Menambahkan dan menghapus element dan node

# Event Handling



# Definisi

Merepresentasikan sebuah kejadian yang nanti akan terjadi di dalam DOM

# Cara mendengarkan Event

## Event Handler

Inline HTML attribute

Element Method

**addEventListener ( )**

# Jenis-jenis Events

Mouse Event

Keyboard Event

Resources Event

Focus Event

View Event

Form Event

CSS Animation & Transition Event

Drag & Drop Event

etc

# DOM Traversal