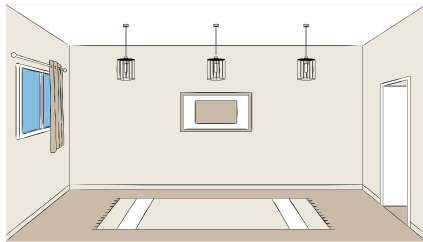


# The 3 Musketeers of Web Dev

**HTML**  
(structure)



**CSS**  
(style)



**JS**  
(logic)



## Starter Code

`<style>` tag connects HTML with CSS

`<script>` tag connects HTML with JS

Apna College

```
<html>  
  <head>  
    <title> Website Name </title>  
  </head>  
  <body>  
    <!-- Content Tags -->  
  </body>  
</html>
```

Apna College

## Window Object

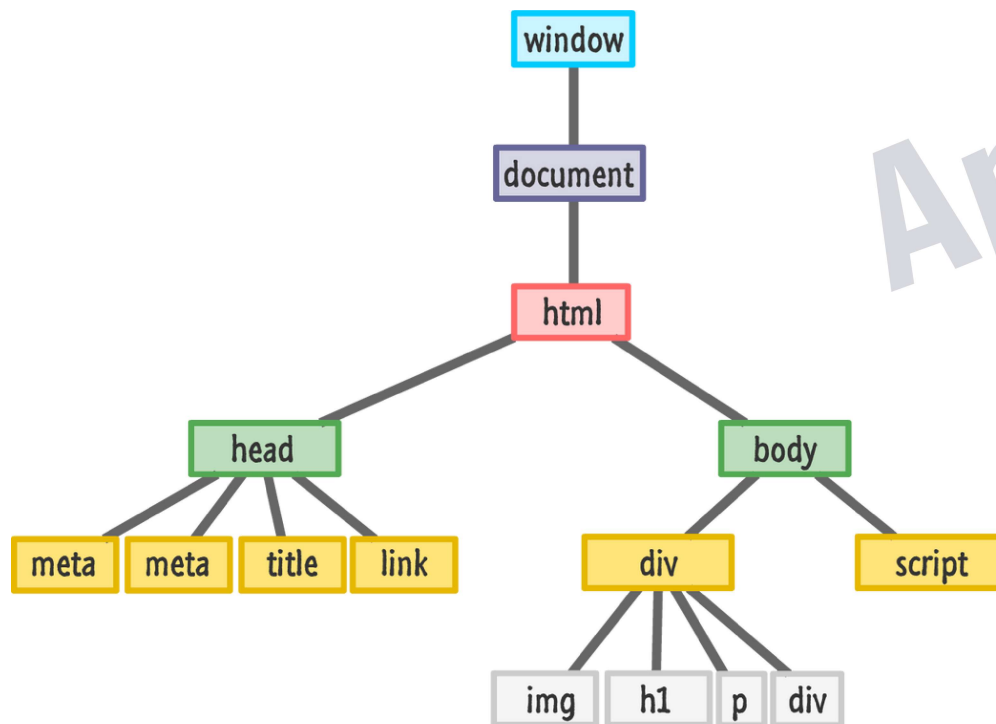
The window object represents an open window in a browser. It is browser's object (not JavaScript's) & is automatically created by browser.

It is a **global** object with lots of properties & methods.

Apna College

# What is DOM?

When a web page is loaded, the browser creates a **Document Object Model (DOM)** of the page



# DOM Manipulation

## Selecting with id

```
document.getElementById("myId")
```

## Selecting with class

```
document.getElementsByClassName("myClass")
```

## Selecting with tag

```
document.getElementsByTagName("p")
```

# DOM Manipulation

---

## Query Selector

```
document.querySelector("#myId / .myClass / tag")
```

//returns first element

```
document.querySelectorAll("#myId / .myClass / tag")
```

//returns a NodeList

# DOM Manipulation

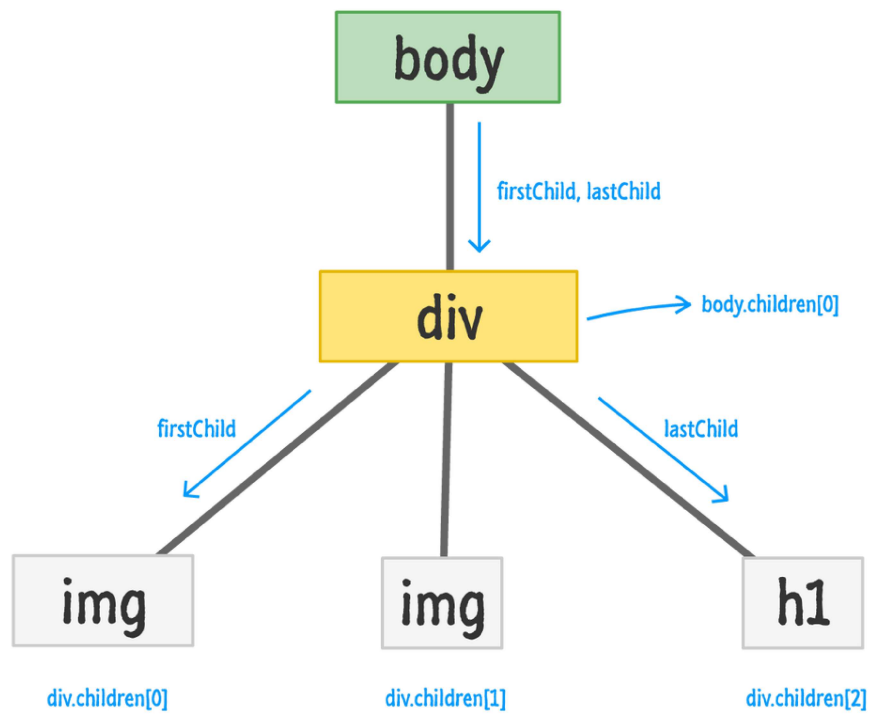
---

## Properties

- **tagName** : returns tag for element nodes
- **innerText** : returns the text content of the element and all its children
- **innerHTML** : returns the plain text or HTML contents in the element
- **textContent** : returns textual content even for hidden elements



# Homework



## Let's Practice

**Qs. Create a H2 heading element with text - “Hello JavaScript”. Append “from Apna College students” to this text using JS.**

**Qs. Create 3 divs with common class name - “box”. Access them & add some unique text to each of them.**

# DOM Manipulation

---

## Attributes

- `getAttribute( attr )` //to get the attribute value
- `setAttribute( attr, value )` //to set the attribute value

## Style

- `node.style`

# DOM Manipulation

---

## Insert Elements

let el = document.createElement("div")

- `node.append( el )` //adds at the end of node (inside)
- `node.prepend( el )` //adds at the start of node (inside)
- `node.before( el )` //adds before the node (outside)
- `node.after( el )` //adds after the node (outside)

## Delete Element

- `node.remove( )` //removes the node

## Let's Practice

**Qs. Create a new button element. Give it a text “click me”, background color of red & text color of white.**

**Insert the button as the first element inside the body tag.**

**Qs. Create a <p> tag in html, give it a class & some styling.**

**Now create a new class in CSS and try to append this class to the <p> element.**

**Did you notice, how you overwrite the class name when you add a new one?  
Solve this problem using classList.**