

Session 1.7

Web Technologies

-

**Understand how  
Internet & Browser  
based apps work**

AN INITIATIVE BY

**UNICAL ACADEMY**

## What will be covered in this session?

- A Business Organization and various functions
- Programs and Projects – from business / users' perspective
- SDLC – V Process model – 2 hours
- Importance of Testing, Quality, and the critical role of a Tester
- Types of Testing – a quick overview

### Part 2: Concepts of Testing (30 h)

- Types of Testing
- Common Testing Tools
- Manual Testing – Test cases, Data, Scenarios etc.
- Case-studies and Scenarios

### Part 3: Automated Testing (50 h)

- Selenium Overview
- Web Driver, Locators, Elements, and more
- Automation and Runs

### Part 4: Hands-on Sessions (100 h)

- Use cases and traceability
- Test data and scenarios
- Day to day work
- Defect prevention, RCA and other value add aspects

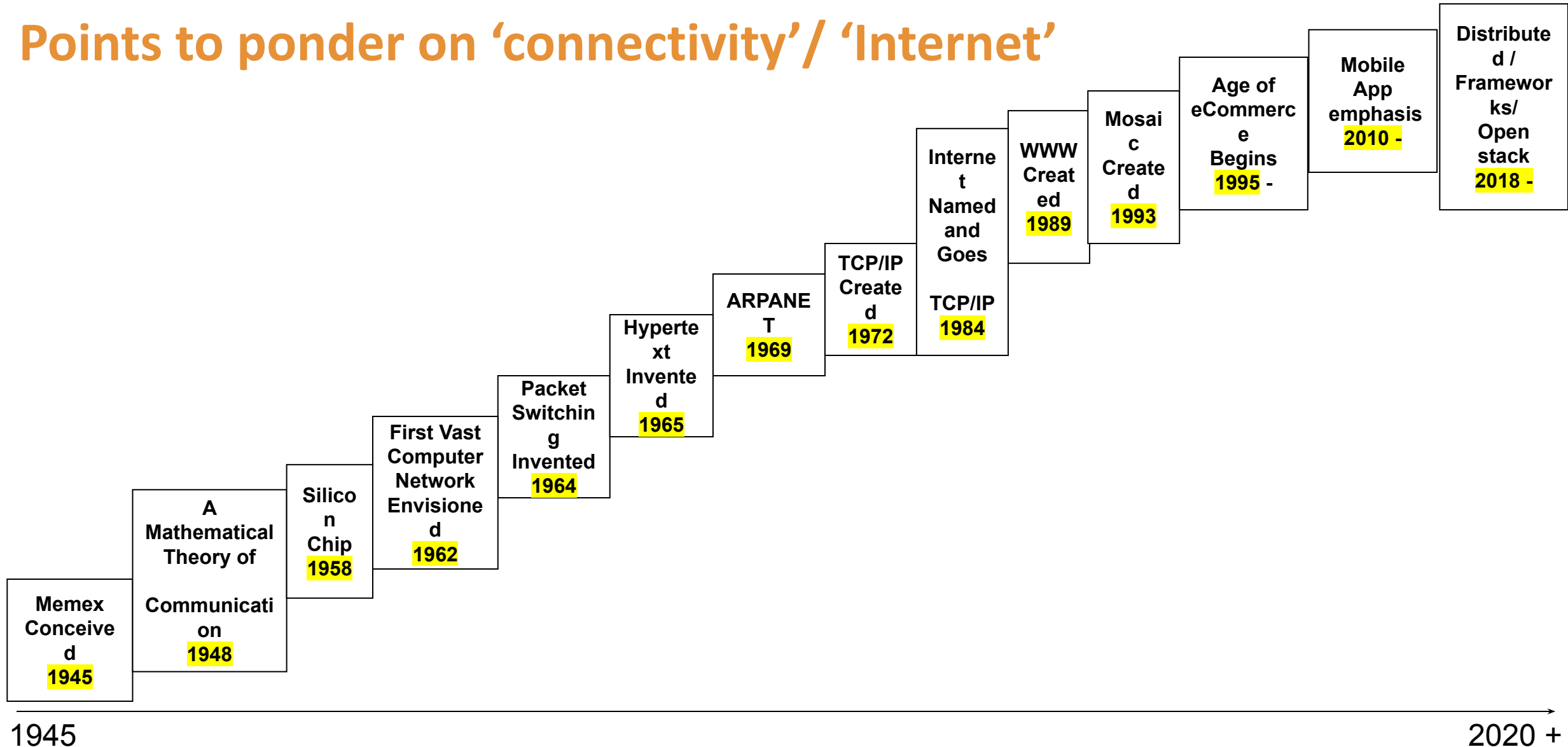
### Part 1: The Basics (20 hours)

- Organization & its working
- SDLC & STLC Overview
- Basics of OOPS, Database Java essentials for Testing
- Overview on few Testing roles Job Descriptions



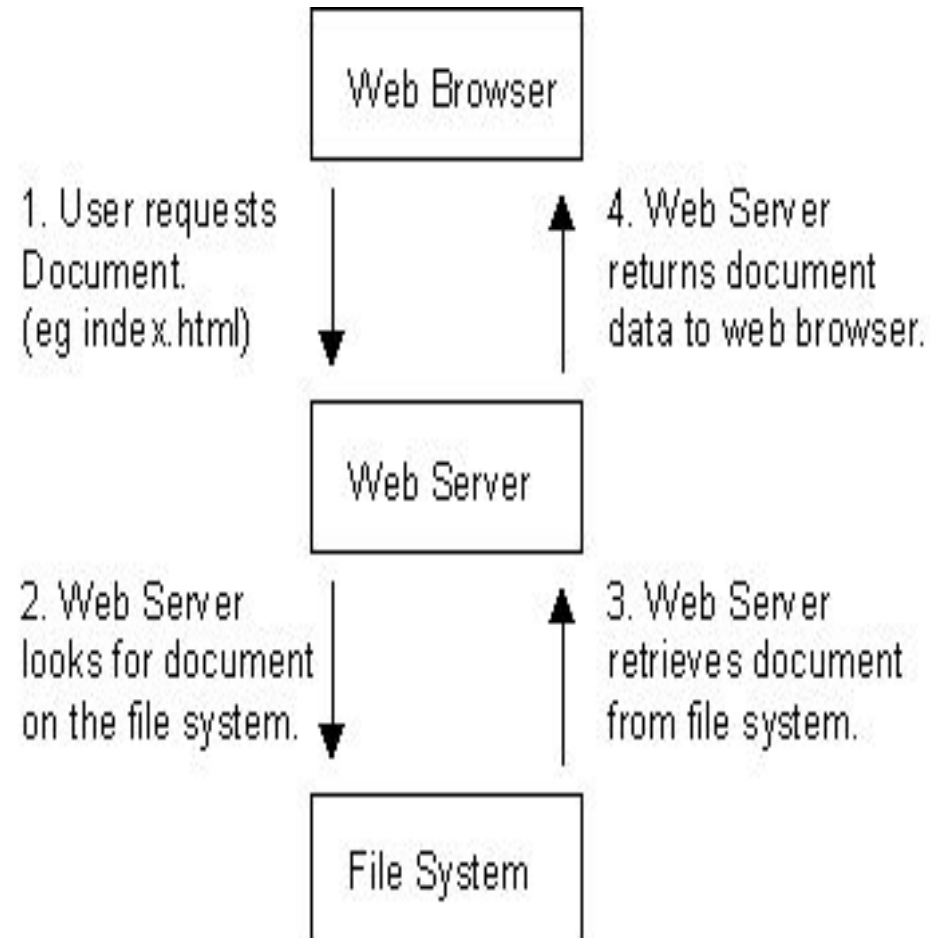
Let's go!!!

# Points to ponder on 'connectivity' / 'Internet'



1945

2020 +



- ★ [HTTP](#) is the protocol that allows Web browsers and servers to communicate
- ★ HTTP/1.0 was the original version
- ★ HTTP/1.1 had an advantage of Persistent Connection
- ★ HTTP/2.0 Server side events and Web sockets – made a difference!

## The Browsers!

Since Selenium deals with Elements and Controls in a Browser, it is worth to learn how the browsers work. This section focuses on browsers, HTTP protocols, HTTP requests, and other related key aspects.

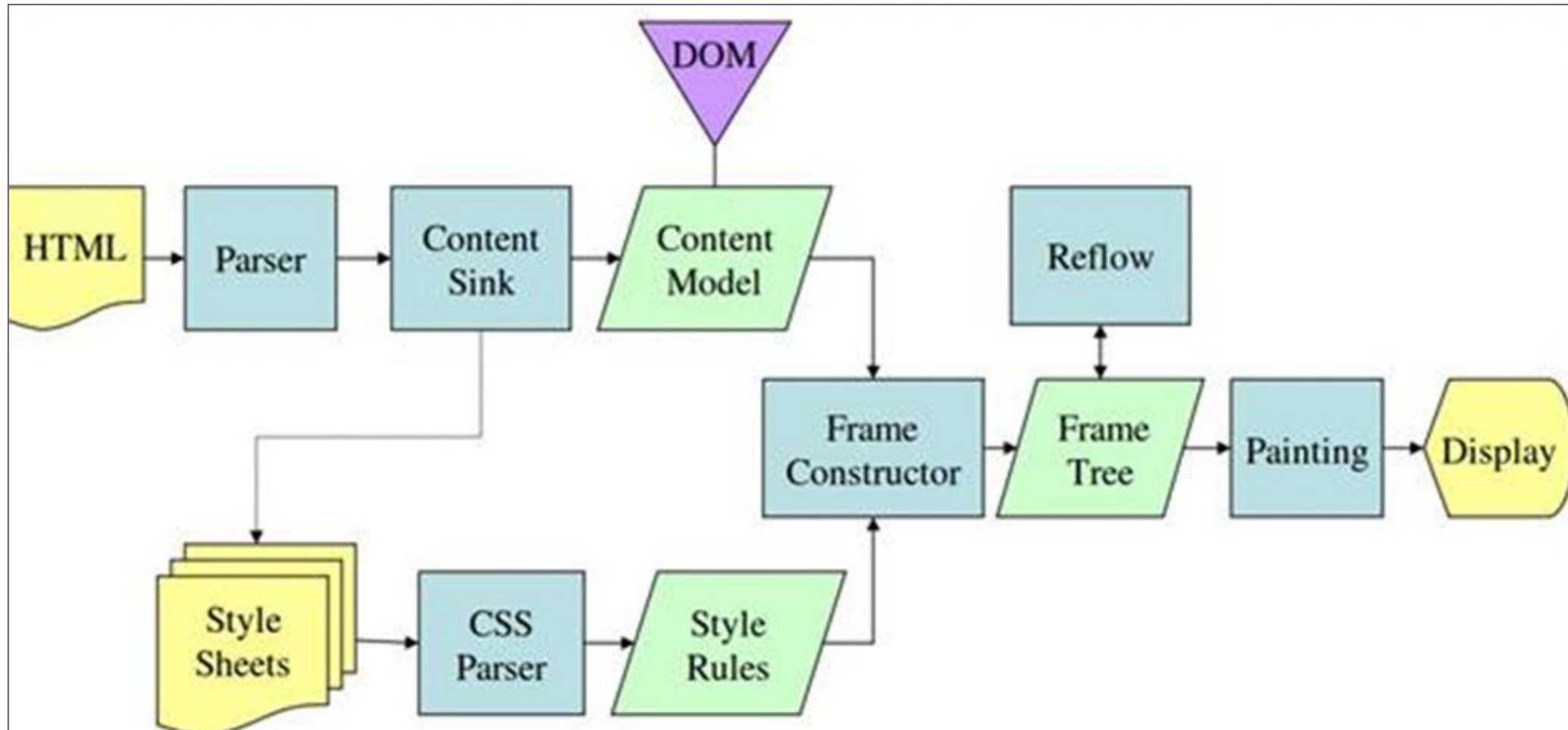
- There are five major browsers on Desktop : Chrome, Firefox, Internet Explorer, Safari and Opera.
- Chrome, Safari, and Firefox make up around 70% of global browser usage
- The main purpose of a browser is to present a web resource that you asked for through its URI (Uniform Resource Identifier). Most of the times the asked web resources are HTML documents / pages.
- Browser interprets a HTML document rendering as per W3C HTML and CSS specifications.

## High level structure of a Browser

**Following are the main components of a Browser -**

- User Interface: Address bar, back/forward button, Menu etc.
- Rendering Engine: This is responsible for displaying requested content
- Browser Engine: Works as intermediary between User Interface and Rendering Engine
- Network Communicator: This makes network calls HTTP requests (Get, Put etc.)
- JavaScript Interpreters: This used to parse and execute JavaScript code.
- UI Backend: Used to render basic GUI (Buttons, Windows etc.) by utilizing OS interface methods
- Data Storage: This helps browsers to save certain data locally. Example cookies, bookmarks etc.

## Firefox browser's Gecko engine workflow

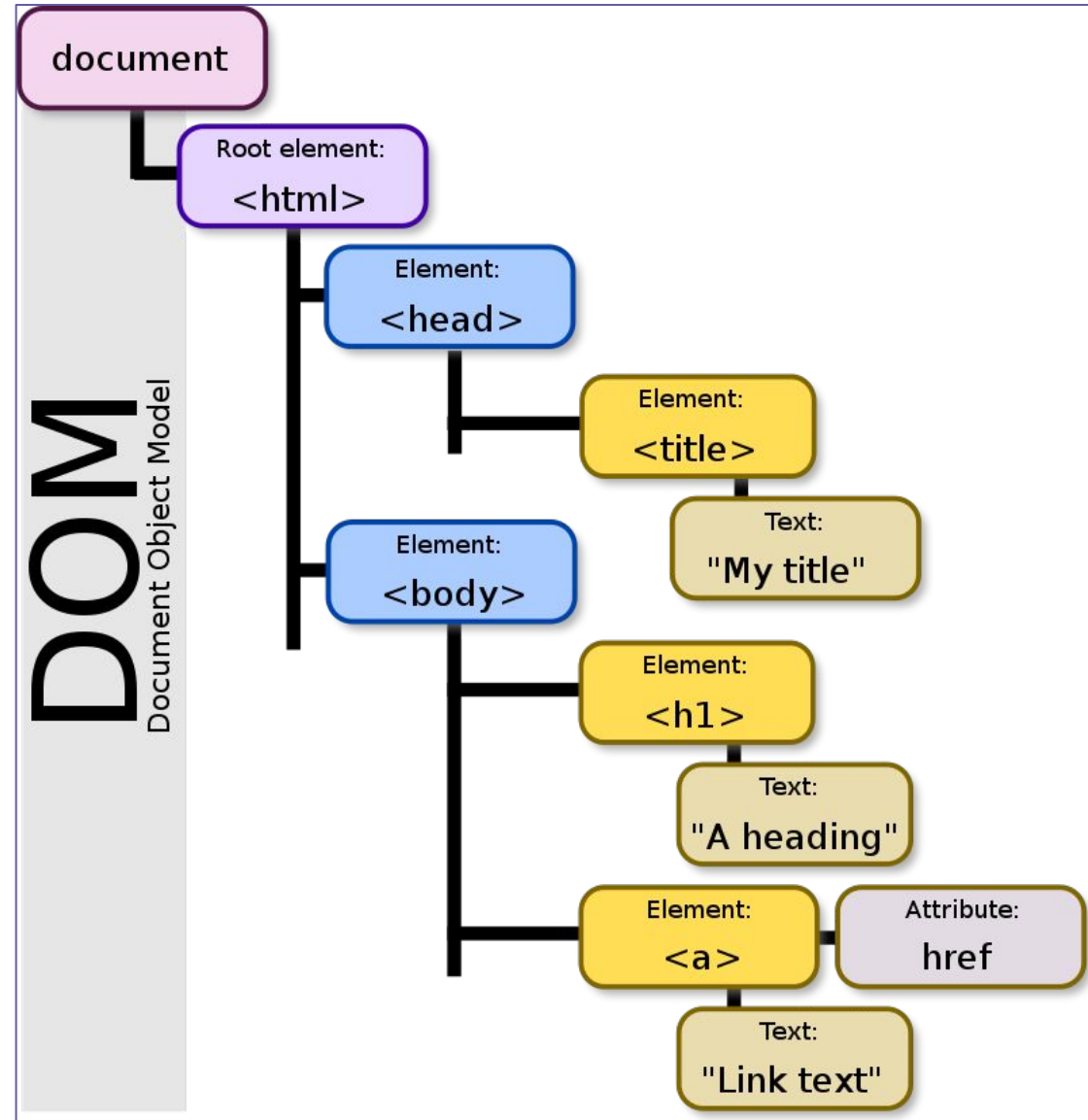


# Document Object Model

Document Object Model (**DOM**) is the way how HTML elements are structured. **Selenium** IDE can use the **DOM** to access page elements.

Four ways to locate HTML Element through DOM:

- getElementById
- getElementsByName
- dom:name
- dom:index





## HTML, CSS, JavaScript – working together

### HTML

**The 'main body' – core content**

Deals with –

- Hypertext Markup Language (HTML)
- Content and basic structure
- Describes and defines
- Made up of tags
- Tells the browser what to display

### CSS

**The 'add-on' / 'accessories' -**

Deals with –

- Cascading Style Sheet (CSS)
- Gives style and structure to the content
- Link the CSS file to the HTML
- Tells the browser how to display

### JavaScript

**The ability to perform actions**

Deals with –

- Behaviour of the website
- Used for interactive functionality
- Allows for the user to interact with the browser

# Integrated View of all 'segments'

