

THE DOM

```
doctype html>
<html lang="en" dir="ltr">
<head>...</head>
<body>
  <div id="toy-header">
    
  </div>
  <div class="container">
    <form class="add-toy-form">
    </form>
  </div>
  <p style="text-align:center">
    "Andy needs your help! "
    <button id="new-toy-btn">Add a new toy!</button>
  </p>
  <div class="toy-collection">
```

DOM

Document Object Model



DOM – Document Object Model

Especifica como os browsers devem criar o modelo, **estrutura de dados de uma página web**

Especifica podemos aceder e manipular conteúdos de uma **página**

O “**Object Model**” porque toda a estrutura é constituído por **objectos**, que têm propriedades e métodos.

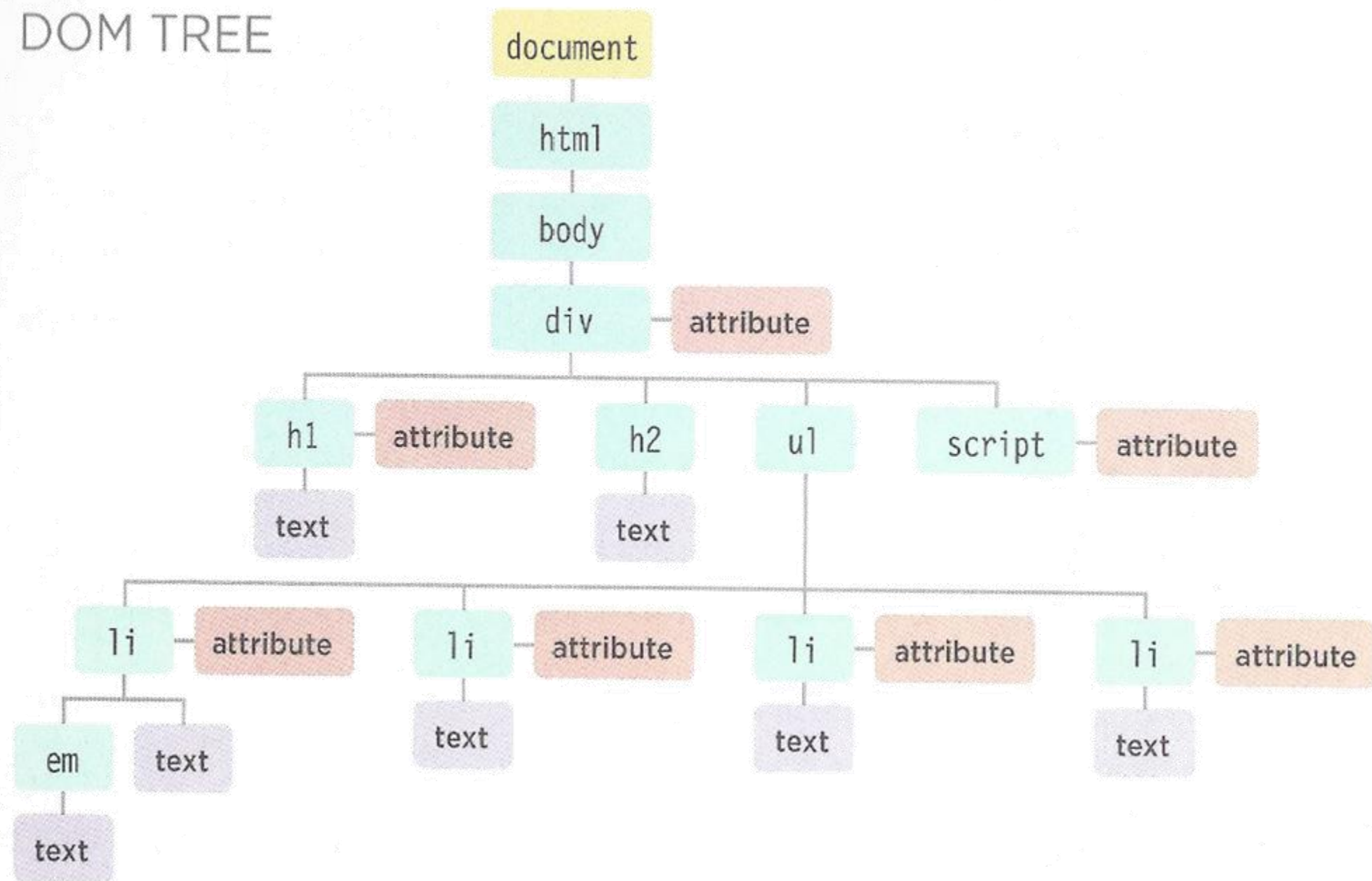


DOM – Árvore

```
<!DOCTYPE html>
<html>
  <head>
    <title>JavaScript : Document Object Model – Get Element By ID</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="css/c05.css">
  </head>
  <body>
    <div id="page">
      <h1 id="header">List King</h1>
      <h2>Buy groceries</h2>
      <ul>
        <li id="one" class="hot"><em>fresh</em> figs</li>
        <li id="two" class="hot">pine nuts</li>
        <li id="three" class="hot">honey</li>
        <li id="four">balsamic vinegar</li>
      </ul>
    </div>
    <script src="js/1_get-element-by-id.js"></script>
  </body>
</html>
```



DOM TREE



DOM

O Modelo do DOM fica na memória do browser e é constituído por “Nodes”

“document node”

“element nodes”

“attribute nodes”

“text nodes”



DOM

Aceder e manipular conteúdos requer 2 passos essenciais:

1. Aceder aos elementos da árvore do DOM
2. Manipular esses elementos



DOM

Aceder aos elementos

Selecionar individualmente cada “Element Node”

Selecionar vários Elementos (NodeLists)

Navegar por “Element Nodes”



DOM

Manipular esses elementos

Alterar/Atualizar conteúdos de “text nodes”

Manipular conteúdos (elementos) de HTML

Aceder/manipular valores de atributos



DOM – aceder a elementos

Através de DOM queries
é devolvido um elemento ou uma “lista de Nodes”
(NodeList) ou colecção de “nodes” (HTMLCollection)

Métodos que retornam um elemento:

`getElementById('id')`

`querySelector('css selector')`



DOM – acessar a elementos

Métodos que retornam uma “nodelist”:

getElementsByTagName('class')
getElementsByTagName('tagname')
querySelectorAll('css selector')



DOM – acceder a elementos

Navegar por elementos (propiedades):

parentNode

previousSibling / nextSibling

firstChild / lastChild



DOM – acceder a elementos

Exemplos



DOM – Manipular conteúdos

nodeValue

innerHTML

textContent

innerText



DOM – manipular elementos e conteúdos

exemplos



DOM – manipular elementos

createElement()
createTextNode()
appendChild()
removeChild()
insertBefore()



DOM – manipular elementos e conteúdos

exemplos



DOM – manipular elementos

getAttribute()

setAttribute()

hasAttribute()

removeAttribute()



DOM – manipular elementos e conteúdos

exemplos



DOM - resumo

O Browser representa a página como uma “tree”

Existem 4 tipos de “nodes”:
document, element, attribute e texto

Podemos selecionar elementos (nodes) pelo “id”, pela “class”, por “tag name”, ou por selector de “css”



DOM - resumo

Ao efectuar uma “DOM Query” é devolvido ou um elemento ou uma colecção (Nodelist ou HTMLCollection)

De um “element node” podemos aceder e alterar conteúdos com propriedades como `textContent` e `innerHTML`

Em browsers mais antigos a implementação do DOM não é consistente (daí a popularidade do JQuery)



Exercicio...

