**DOM模型（一）**

# 授课内容

授课科目：Web开发二；

授课内容：DOM模型（一）；

授课类型：讲授；

授课时间：6学时；

主讲教师：。

# 教学目标要求

## 能力目标：

1. 理解 DOM（文档对象模型）和 DOM 树的概念；
2. 掌握访问文档中指定元素节点的方法；
3. 掌握访问文档中属性结点的方法；
4. 掌握访问文档中的相关节点的方法；
5. 掌握创建新节点的方法；
6. 掌握操作节点的方法；
7. 掌握 innerHTML 属性的使用方法。

## 知识目标：

1. 访问文档中指定元素节点；
2. 访问文档中属性结点；
3. 创建新节点、节点操作。

# 课件分析

概 述：本章是Web开发二的第七章，主要介绍DOM（文档对象模型）和 DOM 树的概念、访问文档中指定的元素节点、访问和设置文档中的属性节点、访问文档中的相关节点（继承关系）、创建新节点、操作节点、innerHTML 属性。

教学重点：访问文档中元素节点的两种不同方式、访问文档中属性结点的方法、在文档中创建新节点、插入结点、删除结点、替换结点的使用、innerHTML 的使用。

教学难点：在文档中创建新节点、插入结点、删除结点、替换结点的使用。

# 教学方法

问题教学法、讲授法、类比法、讨论法等。

# 教学过程

## 课前补充

（0分钟）

## 上节回顾

（0分钟）

## 作业点评

（0分钟）

## 导入新课

（2分钟）



上一章我们学习了BOM模型，JavaScript是通过访问BOM（Browser Object Model）对象来访问、控制、修改客户端(浏览器)。从这章开始，我们来学习一个新的模型——DOM。

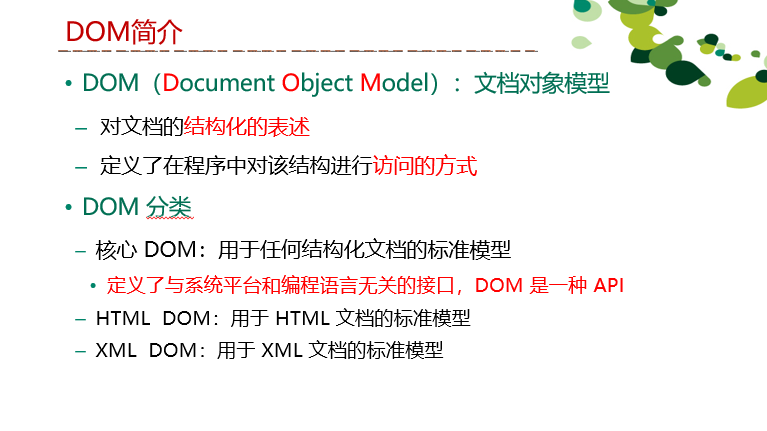
DOM模型和BOM模型是有关联的。BOM的window包含了document，window对象的属性和方法是直接可以使用的，因此可以直接使用window对象的document属性访问、检索、修改XHTML文档内容与结构。而document对象又是DOM模型的根节点。因此可以说，BOM包含了DOM(对象)，浏览器提供出来给予访问的是BOM对象，从BOM对象再访问到DOM对象，从而JavaScript可以操作浏览器以及浏览器读取到的文档。

## 讲授新课

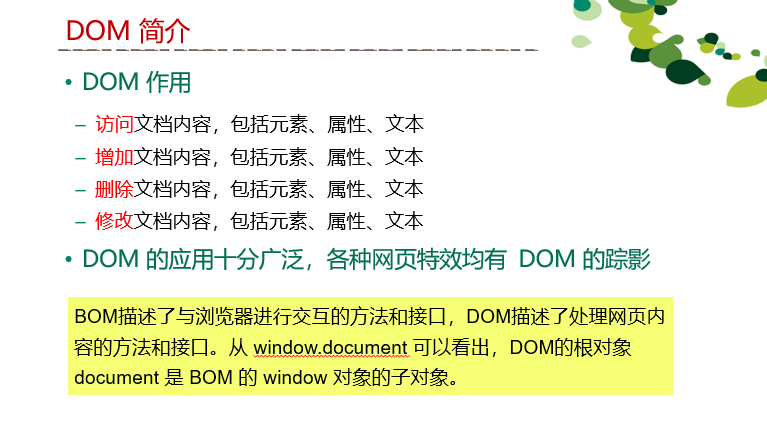
（260分钟）



第一小节学习DOM简介。



DOM是Document Object Model的缩写，中文翻译为文档对象模型。它是对文档的结构化的表述，定义了在程序中对该结构进行访问的方式。DOM可分为三类，在本门课程中主要来学习核心DOM和HTML DOM，定义了与系统平台和编程语言无关的接口。

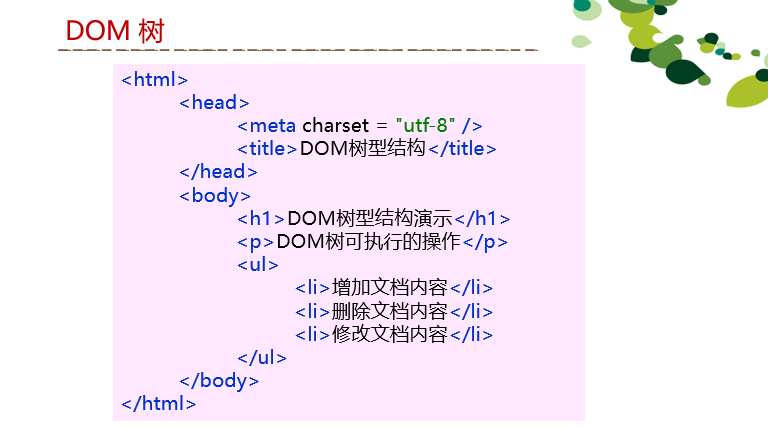


DOM 的应用十分广泛，在各种网页特效中均有 DOM 的踪影，它的主要作用是增加、删除、修改、访问文档内容，包括元素、属性、文本。

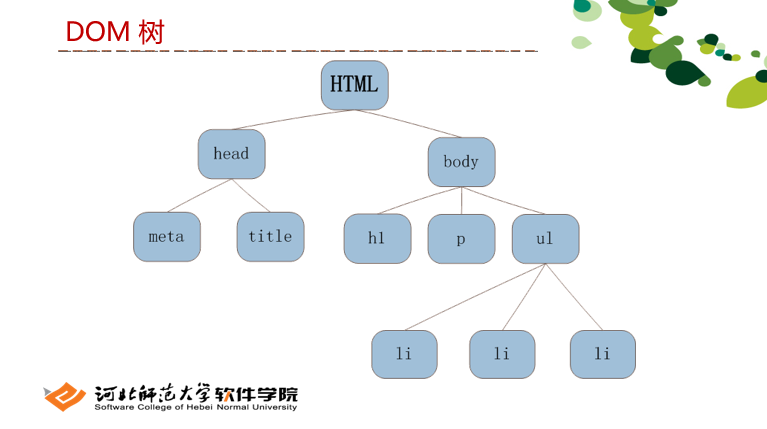
由于BOM的window包含了document，window对象的属性和方法是直接可以使用而且被感知的，因此可以直接使用window对象的document属性，通过document属性就可以访问、检索、修改XHTML文档内容与结构。因为document对象又是DOM模型的根节点。可以说，BOM包含了DOM(对象)，浏览器提供出来给予访问的是BOM对象，从BOM对象再访问到DOM对象，从而js可以操作浏览器以及浏览器读取到的文档。



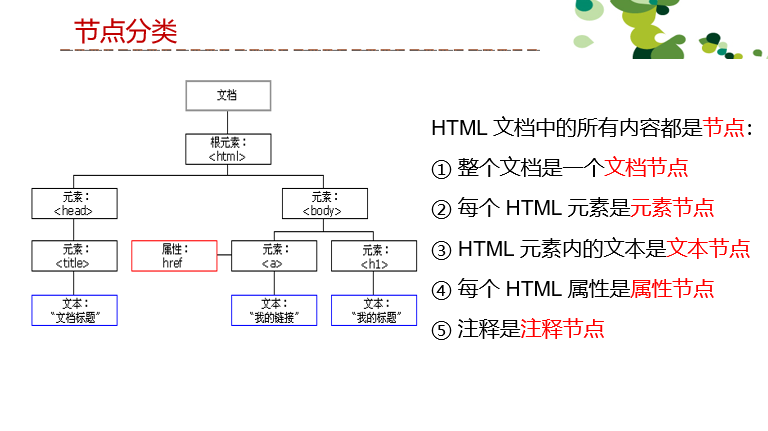
在学习了DOM的基本概念之后，接下来学习DOM树和DOM节点。



这是一个HTML文档的基本结构。

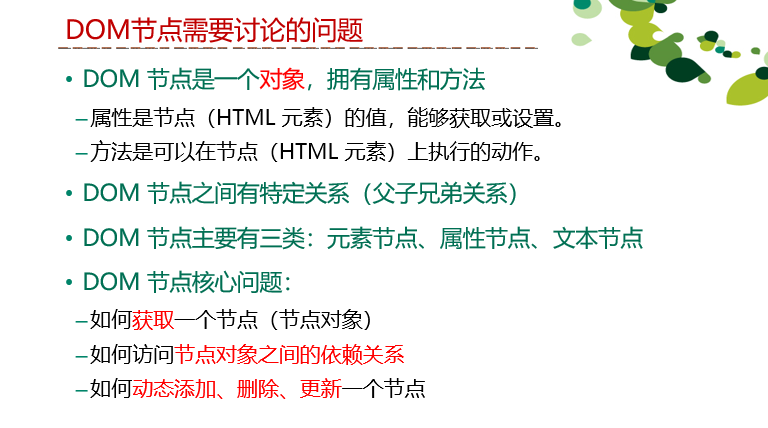


针对文件的结构，可以将其抽象为一棵DOM树。这里可以带着学生一起来画一下DOM树。

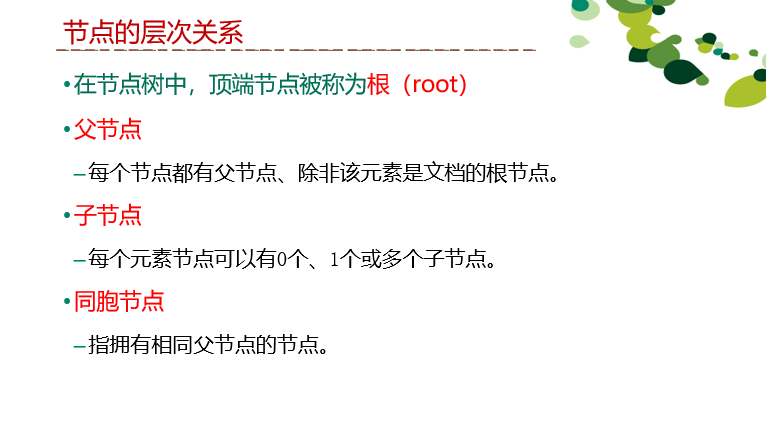


HTML 文档中的所有内容都是节点： 整个文档是一个文档节点，每个 HTML 元素是元素节点，HTML 元素内的文本是文本节点，每个 HTML 属性是属性节点，注释是注释节点。

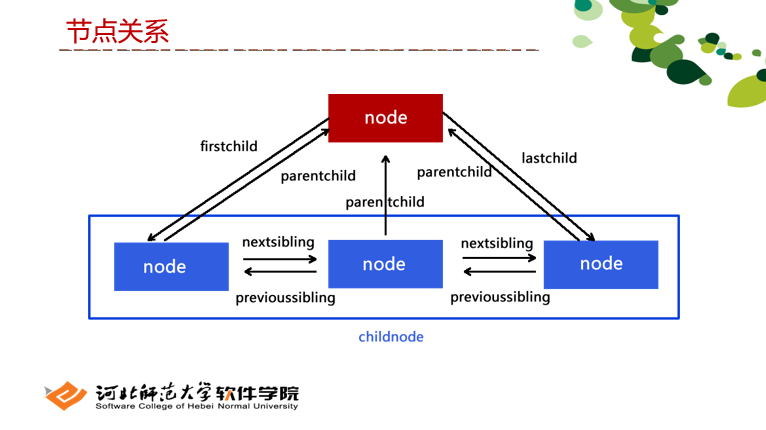
通过这个对象模型，JavaScript 获得创建动态 HTML 的所有力量：JavaScript 能改变页面中的所有 HTML 元素，能改变页面中的所有 HTML 属性，能改变页面中的所有 CSS 样式，能删除已有的 HTML 元素和属性，能添加新的 HTML 元素和属性，能对页面中所有已有的 HTML 事件作出反应，能在页面中创建新的 HTML 事件。



DOM 节点主要有三类：元素节点、属性节点、文本节点。DOM 节点是一个对象，拥有属性和方法。属性是节点（HTML 元素）的值，能够获取或设置，方法是可以在节点（HTML 元素）上执行的动作。DOM 节点之间有特定关系（父子兄弟关系）



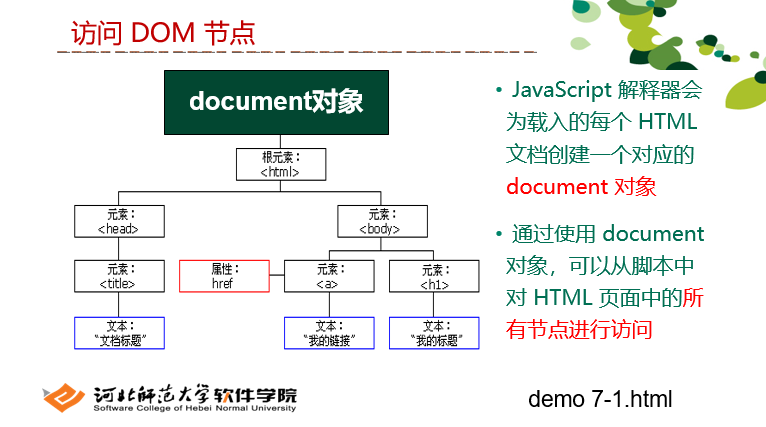
在节点树中，顶端节点被称为根（root）。每个节点都有父节点、除非该元素是文档的根节点。每个元素节点可以有0个、1个或多个子节点。拥有相同父节点的节点成为同胞节点。



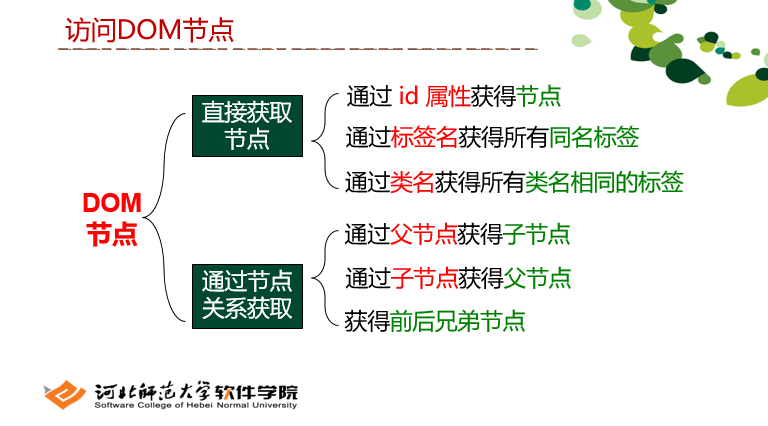
以上几种节点间的关系可以从节点关系图中清晰地看出。红色节点是三个蓝色节点的父节点。三个蓝色节点是红色节点的子节点。三个蓝色节点互为兄弟节点，其中兄弟节点按照先后顺序又分前序兄弟节点和后序兄弟节点。



最后，我们一起来学习如何访问DOM节点，这也是本章的重点内容。

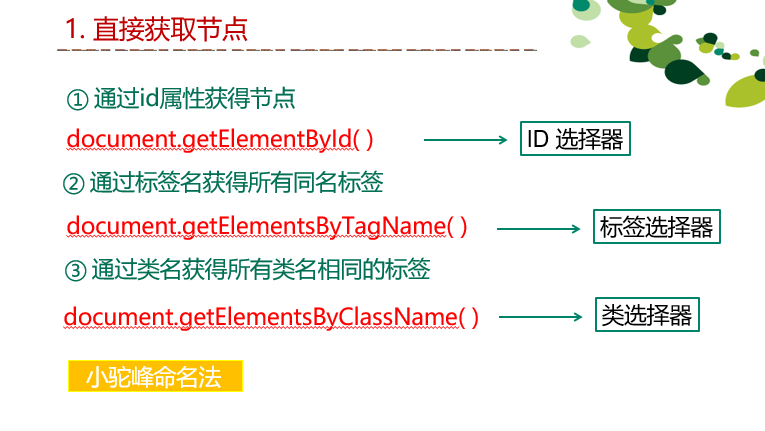


JavaScript 解释器会为载入的每个 HTML文档创建一个对应的document 对象。通过使用 document对象，可以从脚本中对 HTML 页面中的所有节点进行访问。



通常，通过 JavaScript需要操作 HTML 元素。为了做到这件事情，就必须首先找到该元素。访问DOM节点（也叫做获取DOM节点）的方法可以分为两大类，一类是直接获取节点，一类是通过节点之间的关系获取节点。

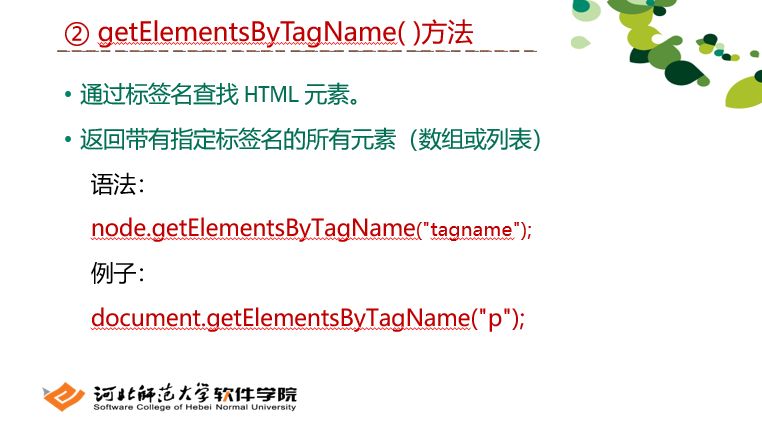
首先，我们来学习直接获取节点的三种方法：通过 id 找到 HTML 元素 ；通过标签名找到 HTML 元素 ；通过类名找到 HTML 元素。



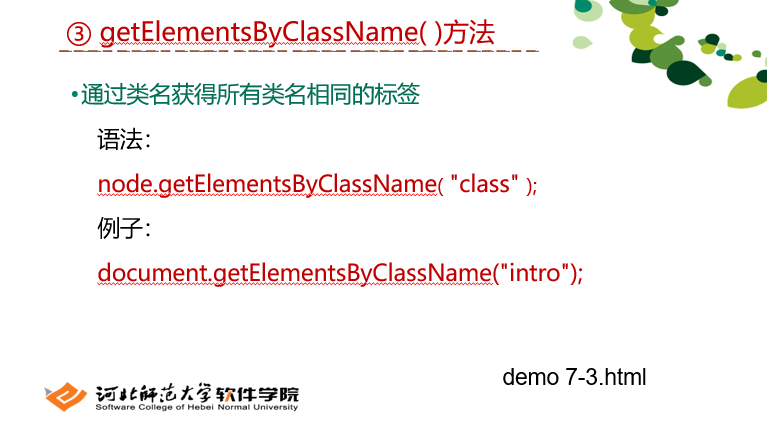
直接获取节点，DOM 中查找 HTML 元素最简单的方法是，使用元素的 id。



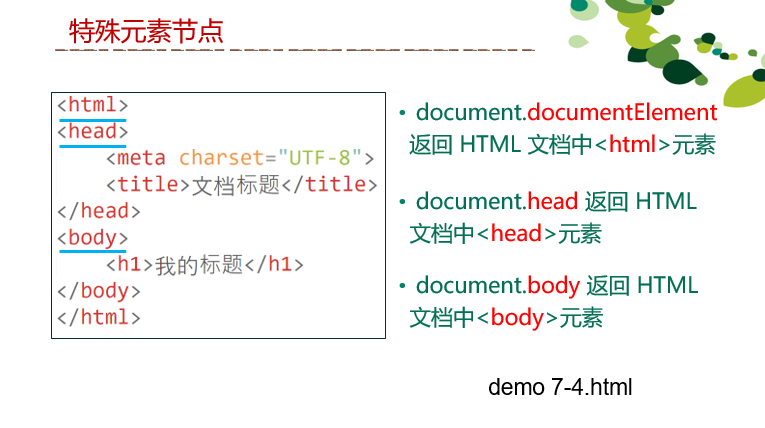
如要查找 id="intro" 的元素：var myElement = document.getElementById("intro"); 如果元素被找到，此方法会以对象返回该元素；如果未找到元素，返回 null。



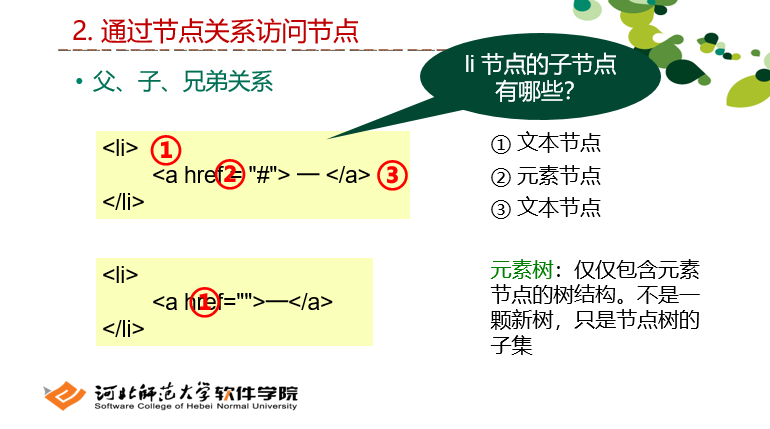
通过标签名查找 HTML 元素，返回带有指定标签名的所有元素（数组或列表）。返回值具有长度length属性。



如果需要找到拥有相同类名的所有 HTML 元素，使用 getElementsByClassName()。如返回包含 class="intro" 的所有元素的列表：var x = document.getElementsByClassName("intro");

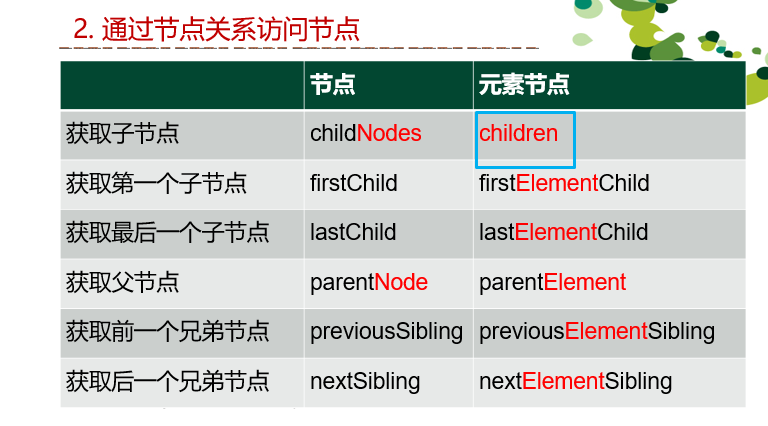


针对HTML文档基本结构中的<html>、<head>、<body>三个特殊元素节点，除了以上三种方法可以直接访问之外，还可以通过document.documentElement、document.head、document.body三种方式来获取，这里需要特别记一下。



接下来，我们来看第二大类，通过节点之间的关系获取节点。

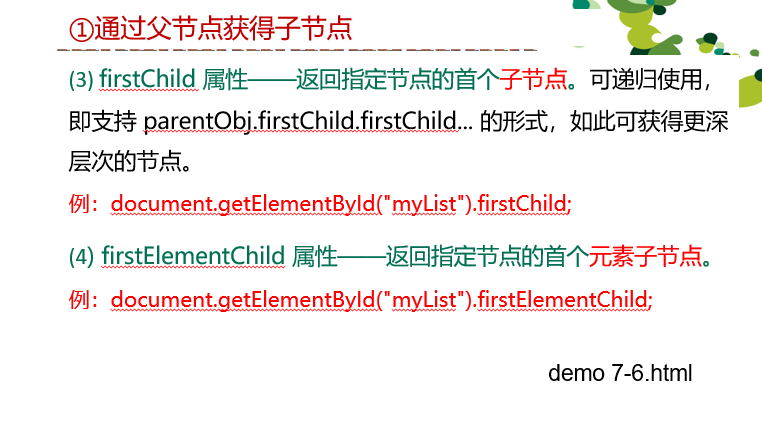
首先，大家一起来说，PPT中li节点的子节点有哪些？① 文本节点——指的是回车。② 元素节点——<a>。③ 文本节点——回车。因此，我们可以发现一个问题，就是通过父节点获取全部子节点时，其中有一些并不是我们所需要的，比如回车。因此，如果仅仅是想要元素节点，而不要其他类型的节点，我们在DOM树的基础上抽象出了一棵新树——元素树。元素树仅仅包含元素节点的树结构，不是一颗新树，只是节点树的子集。



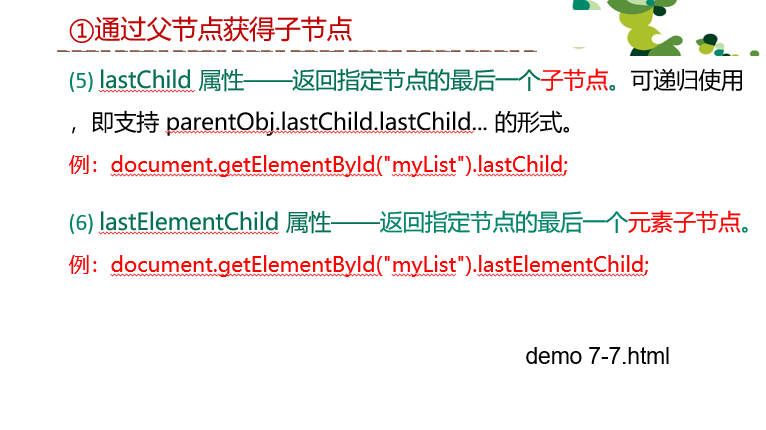
因此通过节点关系访问节点的方法可以分为两大类，PPT上的第一列指的是获取DOM树中的所有类型节点，第二列指的是获取元素树中的元素节点。



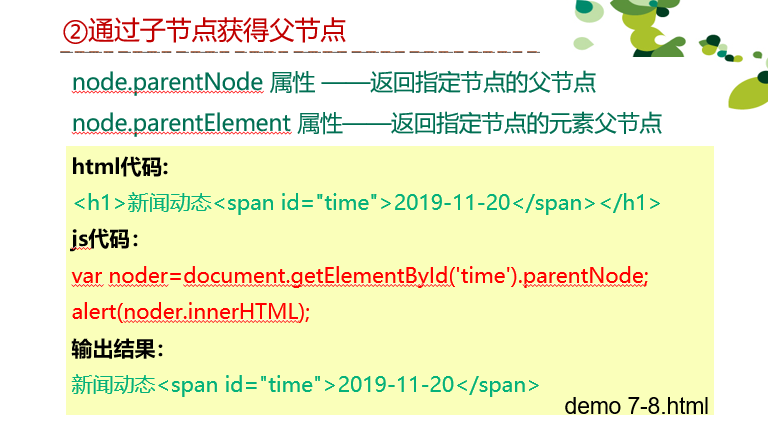
①通过父节点获得子节点。childNodes：它是标准属性，它返回指定元素的子元素集合，包括HTML节点，所有属性，文本节点。children：非标准属性，它返回指定元素的子元素集合。之后均可以通过循环或者索引找到需要的节点。



firstChild 属性——返回指定节点的首个子节点，可递归使用。firstElementChild 属性——返回指定节点的首个元素子节点。

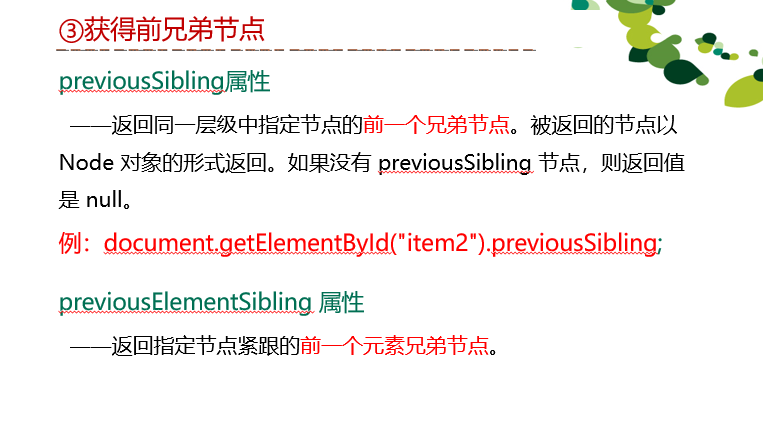


lastChild 属性——返回指定节点的最后一个子节点，可递归使用。lastElementChild 属性——返回指定节点的最后一个元素子节点。

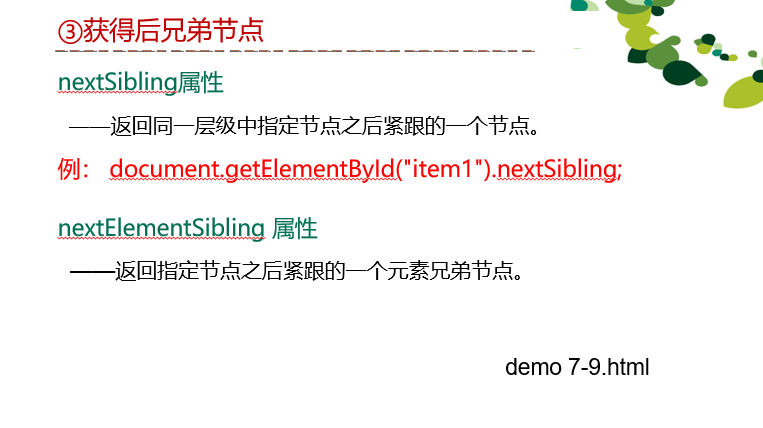


②通过子节点获得父节点。node.parentNode 属性 ——返回指定节点的父节点；

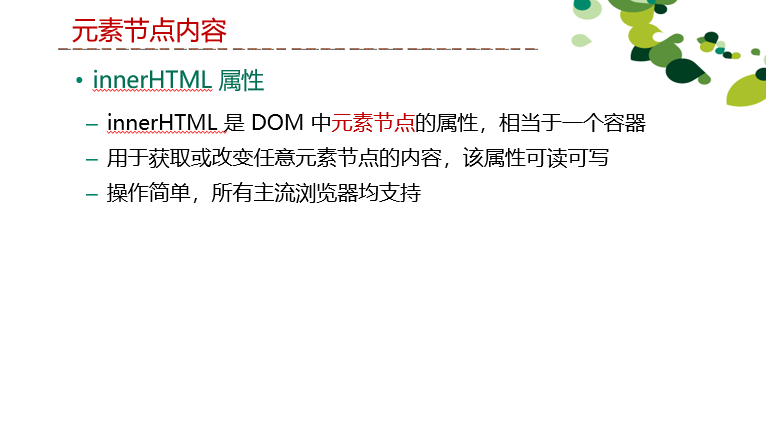
node.parentElement 属性——返回指定节点的元素父节点。



③获得前、后兄弟节点。previousSibling属性——返回同一层级中指定节点的前一个兄弟节点。previousElementSibling 属性——返回指定节点紧跟的前一个元素兄弟节点。被返回的节点以 Node 对象的形式返回。如果没有 previousSibling节点或nextSibling节点，则返回值是 null。



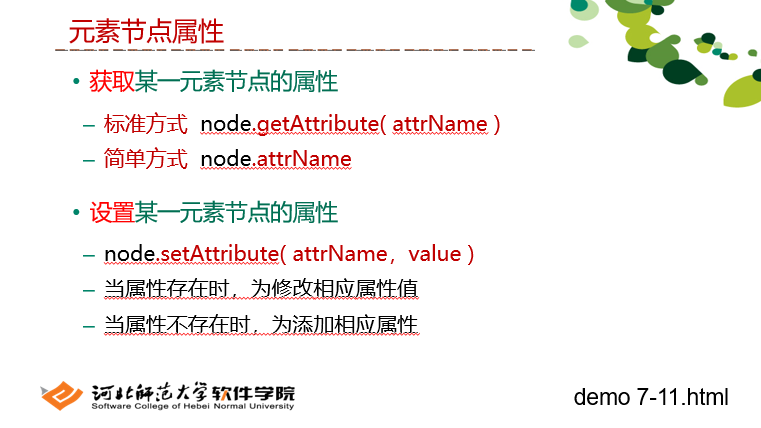
nextSibling属性——返回同一层级中指定节点之后紧跟的一个节点。nextElementSibling 属性——返回指定节点之后紧跟的一个元素兄弟节点。



innerHTML 是 DOM 中元素节点的属性，相当于一个容器用于获取或改变任意元素节点的内容，该属性可读可写。修改 HTML 文档内容最简单的方法是，使用 innerHTML 属性。

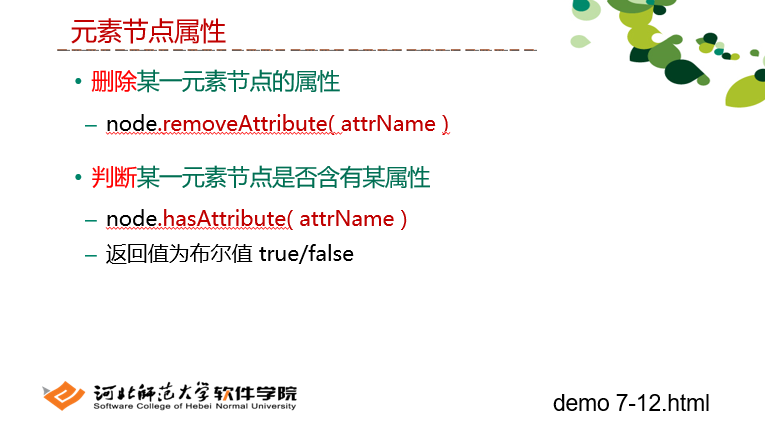
如需修改 HTML 元素的内容，使用此语法：document.getElementById(id).innerHTML = new text;

如需修改 HTML 属性的值，使用如下语法：document.getElementById(id).attribute = new value;



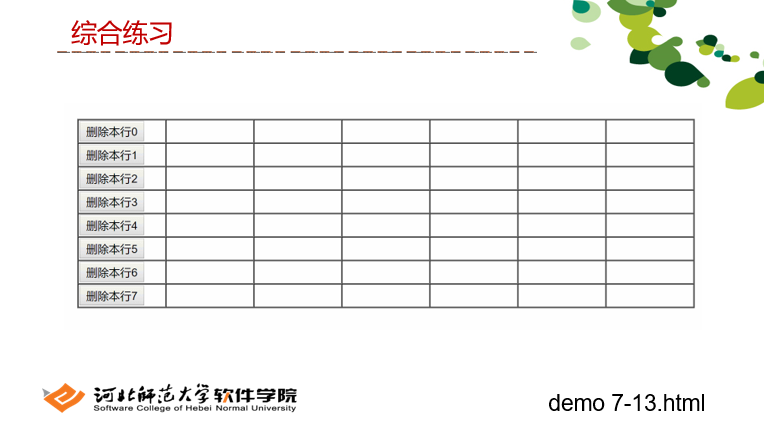
node.getAttribute(someAttribute)获取的是attribute，而node.someAttribute获取的是元素的property，二者并不相同。在大多数情况下property和attribute是同步的，但node.property的方式不能获取自定义属性，node.getAttribute()的方式可以获取自定义属性。因此如果只是想获取非自定义的属性，比如 id、name、src、href 、checked… 用 property 的方式比较符合日常习惯，如果需要获取自定义属性那只能使用 getAttribute。

设置某一元素节点的属性使用node.setAttribute( attrName，value );当属性存在时，为修改相应属性值；当属性不存在时，为添加相应属性。



删除某一元素节点的属性使用node.removeAttribute( attrName );

判断某一元素节点是否含有某属性使用node.hasAttribute( attrName );



使用本章所学知识来实现一个综合练习，要求单击表格第一列单元格中的删除按钮，能够删除单元格所在的一整行。练习中涉及到节点的获取、节点的删除等。



最后我们来进行一下总结。Document的方法 getElementById()返回一个匹配特定 ID的元素。由于元素的 ID 在大部分情况下要求是独一无二的，这个方法自然而然地成为了一个高效查找特定元素的方法。返回一个匹配到 ID 的 DOM Element 对象。若在当前 Document 下没有找到，则返回 null。

getElementsByTagName() 方法可返回带有指定标签名的对象的集合。返回元素的顺序是它们在文档中的顺序。

getElementsByClassName() 方法返回文档中所有指定类名的元素集合，作为 NodeList 对象，它是一个有顺序的节点列表。NodeList 对象可通过节点列表中的节点索引号来访问列表中的节点(索引号由0开始)。



childNodes 属性返回包含被选节点的子节点的 NodeList。parentNode 属性可返回某节点的父节点。如果指定的节点没有父节点则返回 null 。previousSibling 属性返回同一树层级中指定节点的前一个节点。被返回的节点以 Node 对象的形式返回，如果没有 previousSibling 节点，则返回值是 null。getAttributeNode() 方法从当前元素中通过名称获取属性节点。

innerHTML 属性用于获取或替换 HTML 元素的内容。注意书写，innerHTML区分大小写。

## 布置作业

（1分钟）

1. 完成本次课所对应的雪梨作业。