### 任务3：制作复选框全选效果

#### 3.1 document对象的常用方法

##### 1. getElementById方法

此方法的唯一参数是带引号的字符串，该字符串包含要引用的元素ID，如下所示：

document.getElementById(“id”)

该方法返回一个值，通常将该值保存在一个变量中，供随后的脚本语句使用，例如：

var btn = document.getElementById(“btnId”);

在赋值语句后，变量就表示元素对象，可用于获取和设置该对象的属性，或者调用此类对象的方法。

##### 2. document.getElementsByName()方法

可以使用getElementsByName()方法方便地收集一组name属性相同的页面元素。

##### 3. document.getElementsByTagName()方法

可以使用getElementsByTagName()方法方便地收集一组同一个标记名称的页面元素。例如，要获得页面上所有图像，可以调用：

var allImages = document.getElementsByTagName(‘img’);

使用getElementsByTagName的一个优点是，它可以在页面的任何容器元素中收集某个集合。

##### 4. document.createElement()和document.createTextNode()方法

在HTML文档中添加新元素至少需要两个步骤：

(1)为文档创建新元素。

(2)在页面的树形结构中，将其插入到需要的位置。

Document.createElement方法可以在浏览器的内存中创建一个全新的元素对象。指定要创建的元素时，应将元素的标记名作为该方法的字符串参数：

var newElem = document.createElement(“p”);

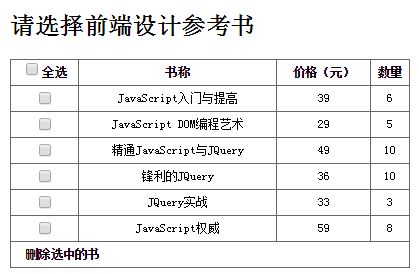
document对象常用的方法如表所示。

|  |  |
| --- | --- |
| 方法 | 说明 |
| close() | 关闭用document.open()方法打开的输出流，并显示选定的数据 |
| getElementById() | 返回对拥有指定id的第一个对象的引用 |
| getElementsByName | 返回带有指定名称的对象集合 |
| getElementsByTagName | 返回带有指定标签名的对象集合 |
| open | 打开一个流，以收集来自任何document.write()或document.write()方法的输出 |
| write | 向文档写HTML表达式或JavaScript代码 |
| writeLn() | 等同于write()方法，不同的是在每个表达式之后写一个换行符 |

#### 3.2 任务实现

#### 3.3 练一练

1.在页面实现全选和全不选的效果，用户勾选全选时，能够将所列书全部选中，全选前的复选框不选中，所列书也将不选中，效果如图所示。



#### 3.4 课后作业

在页面初始状态如图1所示，单击“全选”按钮后，能够实现全选功能，并能将选择内容显示在如图2所示页面，按钮上的文字改为“全不选”；再次单击”全不选“按钮，能实现全不选功能，按钮上的文字改为“全选”

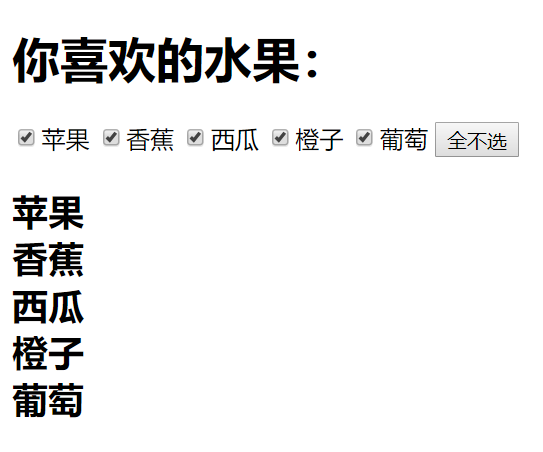
 

图１页面初始状态　　　　　　　　　　　　　　　 图２　全选效果