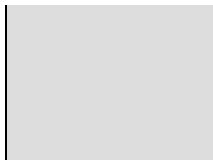
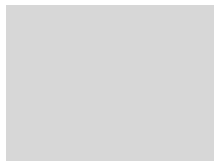
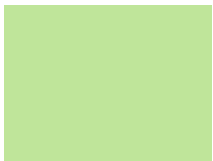
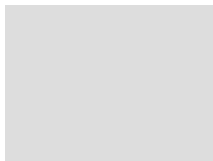


任务3：使用HTML DOM方式动态添加表格



任务3：使用HTML DOM方式动态添加表格



任务描述：



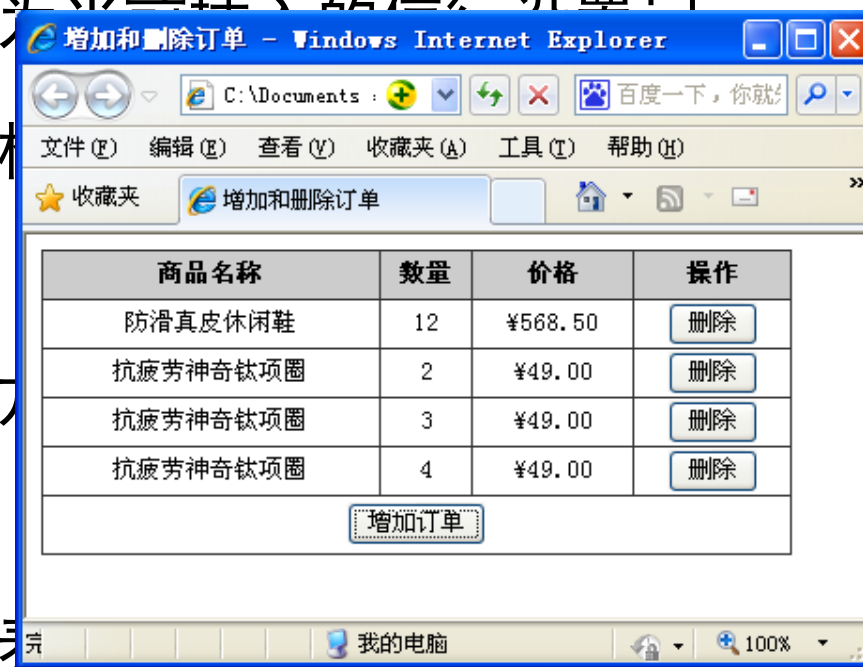
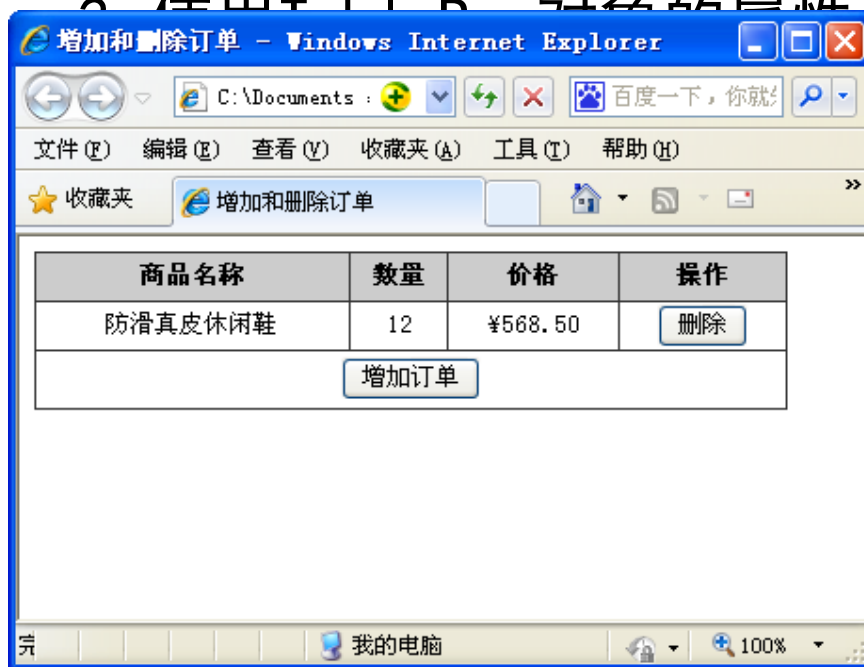
- 单击“增加订单”按钮插入一行订单，当多次单击“增加订单”按钮时，可增加多行。
- 单击“删除”按钮，可删除当前订单。

任务3： 使用HTML DOM方式动态添加表格



任务分析：

1. 使用集合rows和属性length计算当前表格中的行数

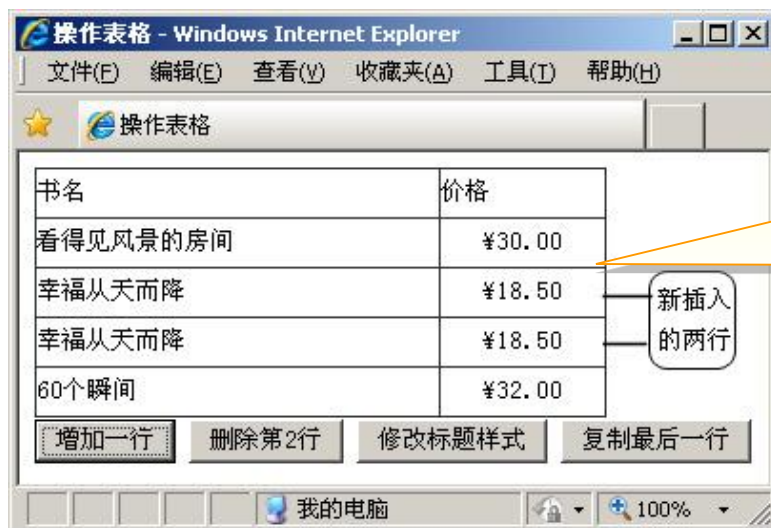


deleteRow() 删除当前行

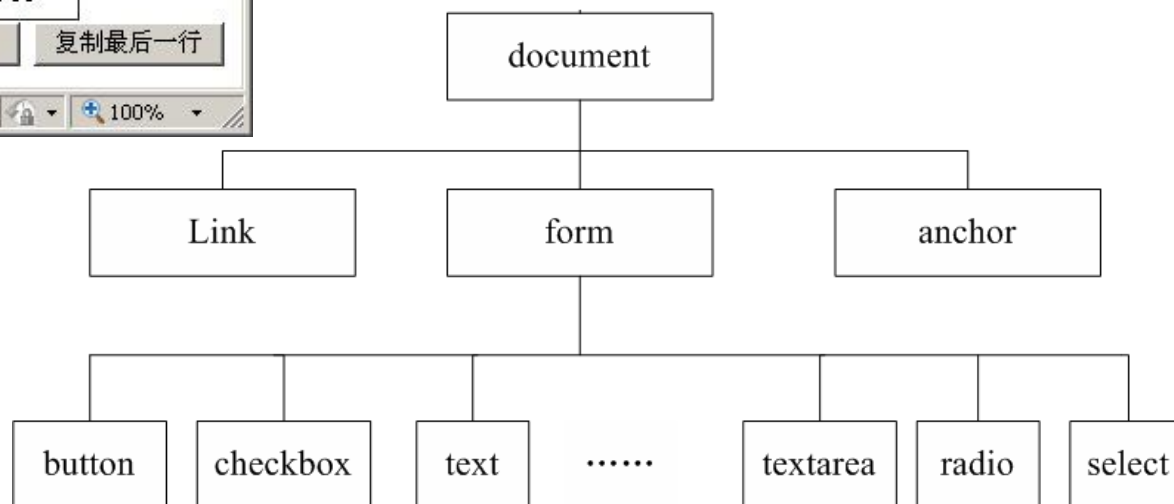
3.1 HTML DOM的特有对象和操作



• 什么是HTML DOM对象



HTML文档中的每个节点都是DOM对象，每类对象都有1套属性、方法。
最常用的是<table>表格和各类表单<form>元素对象



3.1 HTML DOM的特有对象和操作



- HTML DOM对象的属性访问

```
function change(){  
    var imgs=document.getElementById("s1");  
    imgs.src="images/grape.jpg";  
}  
  
function show(){  
    var hText=document.getElementById("s1").alt;  
    alert("图片的alt是: "+hText)  
}  
</script>
```

不再使用**setAttribute ()**
/getAttribute () 方法
简化为对象名.属性值进行
读取或修改



3.2 HTML DOM对象-table对象



- 思路分析

- 1、使用Core DOM标准操作获取表格对象
- 2、使用HTML DOM的table相关对象的属性、方法、事件操作表格



使用HTML
DOM如何实现？

3.2.1 表格相关对象



- 表格相关对象

The screenshot shows a web browser window with the title '使用HTML DOM操作表格'. It contains a table with two columns: '书名' (Book Name) and '价格' (Price). The table has three data rows. A red box highlights the entire table, with a callout pointing to it labeled 'table表格对象'. A green box highlights a cell in the first row, with a callout pointing to it labeled 'tableCell单元格对象'. A blue box highlights the second row, with a callout pointing to it labeled 'tableRow表格行对象'. Below the table are three buttons: '增加一行' (Add a row), '删除第2行' (Delete the 2nd row), and '修改标题' (Modify title).

书名	价格
看得见风景的房间	
幸福从天而降	¥18.5
60个瞬间	¥32.00

增加一行 删除第2行 修改标题

3.2.1 表格相关对象



- table表格对象

类别	名称	描述
属性	<code>rows[]</code>	返回包含表格中所有行的一个数组
方法	<code>insertRow()</code>	在表格中插入一个新行
	<code>deleteRow()</code>	从表格中删除一行

.....代码片段.....

```
tableObject.rows[ ]
```

```
tableObject.insertRow(index)
```

```
deleteRow(index)
```

.....

用法示
例代码

3.2.1 表格相关对象



- tableRow表格行对象

类别	名称	描述
属性	cells[]	返回包含行中所有单元格的一个数组
	rowIndex	返回该行在表中的位置
方法	insertCell()	在一行中的指定位置插入一个空的<td>标签
	deleteCell()	删除行中指定的单元格

.....代码片段.....

```
tableRowObject.insertCell(index)
tableRowObject.deleteRow(index)
```

.....

用法
示例

3.2.1 表格相关对象



- tableCell 单元格行对象

类别	名称	描述
属性	<code>cellIndex</code>	返回单元格在某行单元格集合中的位置
	<code>innerHTML</code>	设置或返回单元格的开始标签和结束标签之间的HTML
	<code>align</code>	设置或返回单元格内部数据的水平排列方式
	<code>className</code>	设置或返回元素的class属性

.....代码片段.....

```
cell2.innerHTML="改变单元格的HTML内容";  
cell2.align="center";
```

.....

用法
示例

3.3 用table对象重新实现表格操作



- 实现“增加一行”的操作

```
function addRow(){  
    var newRow=document.getElementById("myTable").insertRow(-1);  
    var col1=newRow.insertCell(0);  
    col1.innerHTML="幸福从天而降";  
    var col2=newRow.insertCell(1);  
    col2.innerHTML="¥18.5";  
    col2.align="center";  
}
```

增加一个行对象，然后插入单元格，最后设置单元格内容



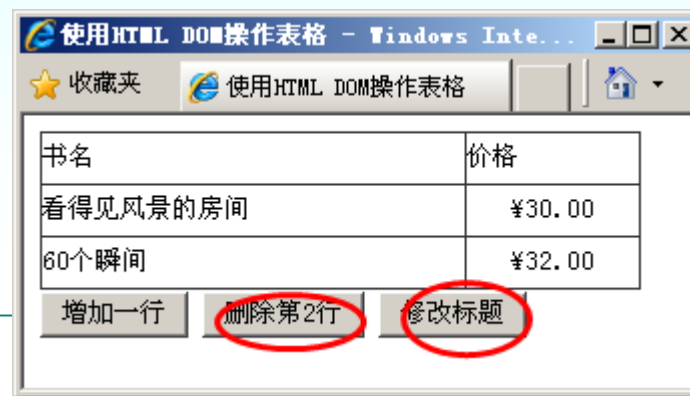
3.3 用table对象重新实现表格操作



- 实现“删除第2行”的操作
- 实现“修改标题”的操作

```
function delRow(){  
    document.getElementById("myTable").deleteRow(1);  
}  
function updateRow(){  
    var uRow=document.getElementById("myTable").rows[0];  
    uRow.className="title";  
}
```

使用**class**样式属性直接修改标题样式



任务实现



```
function addRow(){
    var addTable=document.getElementById("order");
    var row_index=addTable.rows.length-1;    //新插入行在表格中的位置
    var newRow=addTable.insertRow(row_index); //插入新行
    newRow.id="row"+row_index;                //设置新插入行的ID

    var col1=newRow.insertCell(0);
    col1.innerHTML="抗疫劳神奇钛项圈";

    var col2=newRow.insertCell(1);
    col2.innerHTML=row_index;

    var col3=newRow.insertCell(2);
    col3.innerHTML="¥49.00";

    var col4=newRow.insertCell(3);
    col4.innerHTML="<input name='del"+row_index+" ' type='button'
value='删除' onclick=\"delRow('row"+row_index+" ')\" />";    }
```

任务实现



```
function delRow(rowId){  
    var row=document.getElementById(rowId).rowIndex; //删除行所在  
    表格中的位置  
    document.getElementById("order").deleteRow(row);  
}
```



总结



- W3C提供了DOM的哪三类标准？
- Core DOM包括哪些常用的节点操作？方法分别是什么？
- Core DOM、HTML DOM访问属性的方法分别是什么？
- HTML DOM对象-table相关对象包括哪些？分别包含哪些常用的属性、方法？