

第一章

JavaScript基础

■ 学完本门课程后，你能够

实现客户端表单验证

制作网页特效

课程结构图

使用jQuery快速高效制作网页交互特效

第1章 JavaScript基础 (重点)

第2章 JavaScript对象 (重点)

第3章 初识jQuery

第4章 jQuery选择器 (重点)

第5章 jQuery中的事件和动画 (重点)

第6章 使用jQuery操作DOM (重点)

第7章 表单校验 (重点、难点)

第8章 指导学习：课程总复习

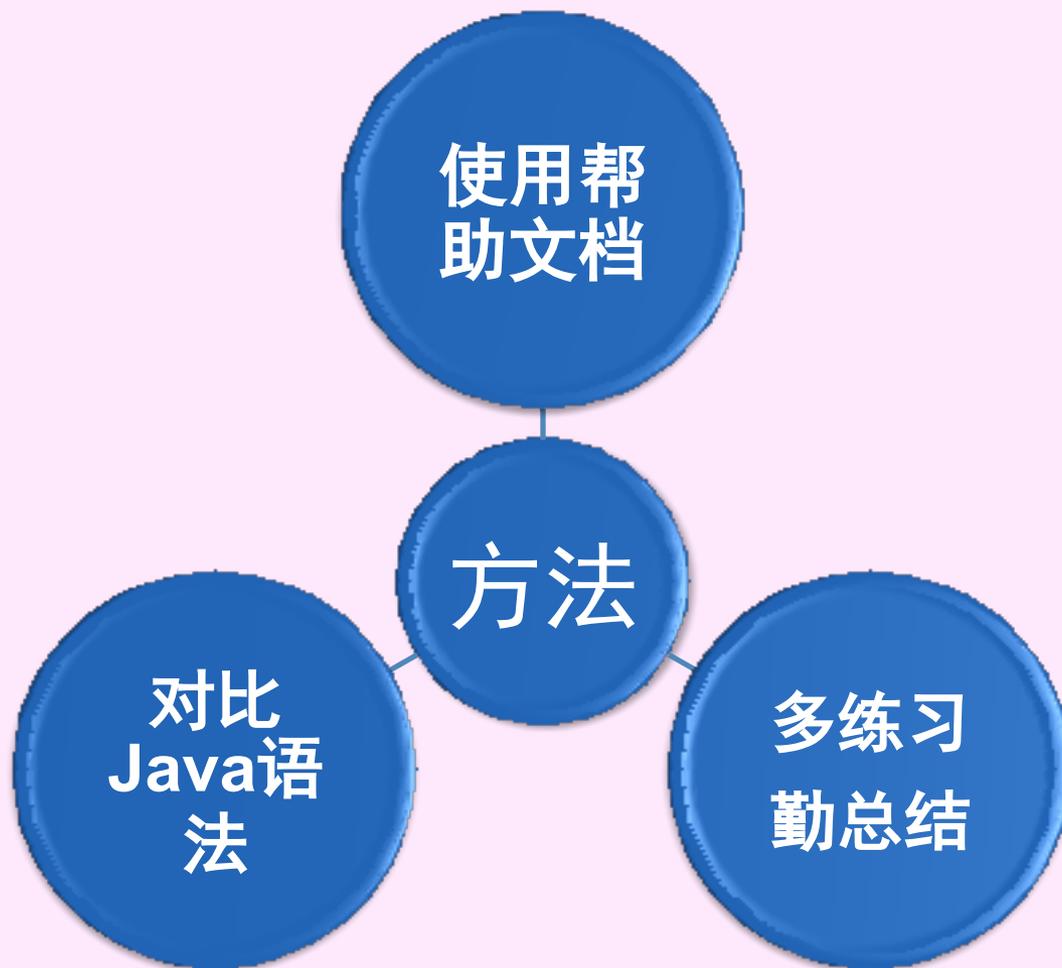
第9章 项目案例：当当网上书店



提问

- 说出Java的选择结构的种类及各自的语法和使用场合
- Java中的循环结构有哪几种？
- 如何定义一个数组并访问？
- 列举字符串常用的方法及其作用。

学习方法





提问

- 请说出网页引用JavaScript的三种方式
- JavaScript常用的基本数据类型有哪些？
- 如何使用JavaScript接收用户输入的信息？
- 如何定义一个函数？

本章目标

- 掌握脚本的基本结构
- 掌握JavaScript的组成
- 掌握JavaScript的基本语法
- 会定义和使用函数
- 会使用工具进行代码调试

重点

重点

重点

难点

为什么要学JavaScript

■ 表单验证—减轻服务器端压力



演示1：注册表单验证

■ 页面动态效果



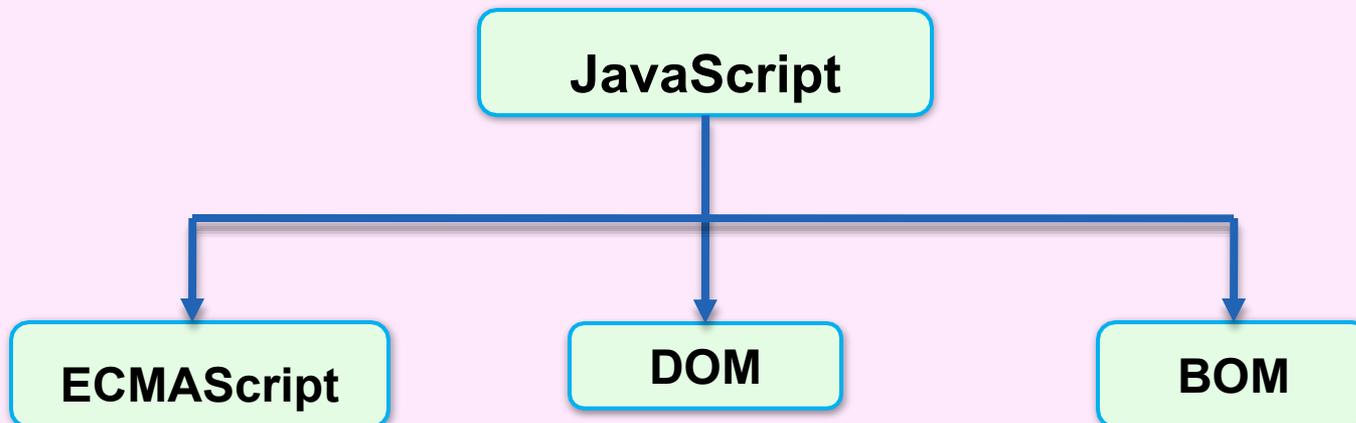
演示2：层的切换



演示2：树形菜单

什么是JavaScript

- JavaScript是一种基于对象和事件驱动的、并具有安全性能的脚本语言。
- JavaScript特点
 - ◆ 向HTML页面中添加交互行为
 - ◆ 脚本语言，语法和Java类似
 - ◆ 解释性语言，边执行边解释
- JavaScript组成



JavaScript的基本结构

■ JavaScript的基本结构

```
<script type="text/javascript">  
  <!--  
    JavaScript 语句;  
  -->  
</script >
```

JavaScript的应用

```
.....  
<title>输出Hello Word!</title>  
<script type="text/javascript">  
  <!--  
    document.write("输出HelloWorld");  
    document.write("<h1>Hello World</h1>");  
  -->  
</script>  
</head>  
<body>页面主体内容</body>  
</html>
```

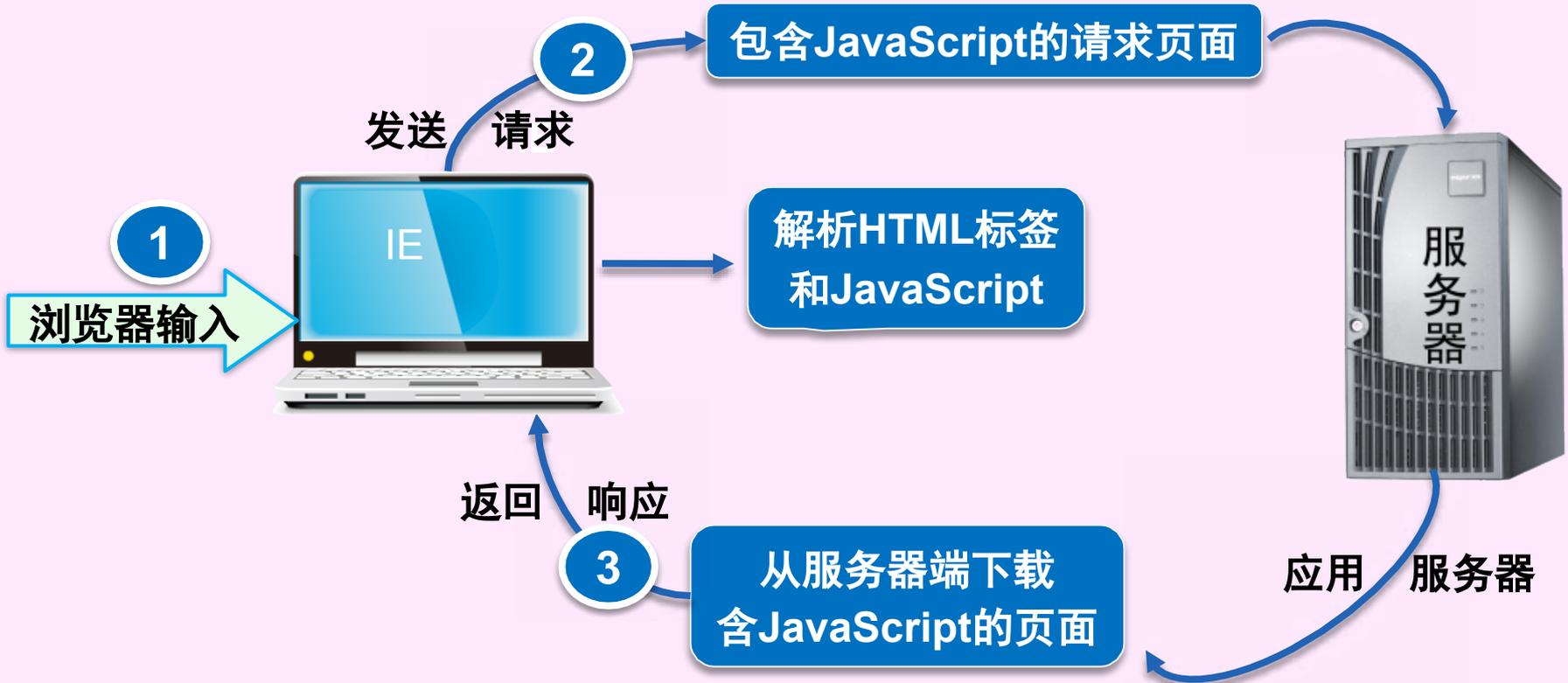
\$经验

`<script>...</script>`可以包含在文档中的任何地方，只要保证这些代码在被使用前已读取并加载到内存即可。



演示示例1：输出Hello World

JavaScript的执行原理



网页中引用JavaScript的方式

- 使用<script>标签
- 外部JS文件

```
<script src="hello.js" language="javascript"></script>
```

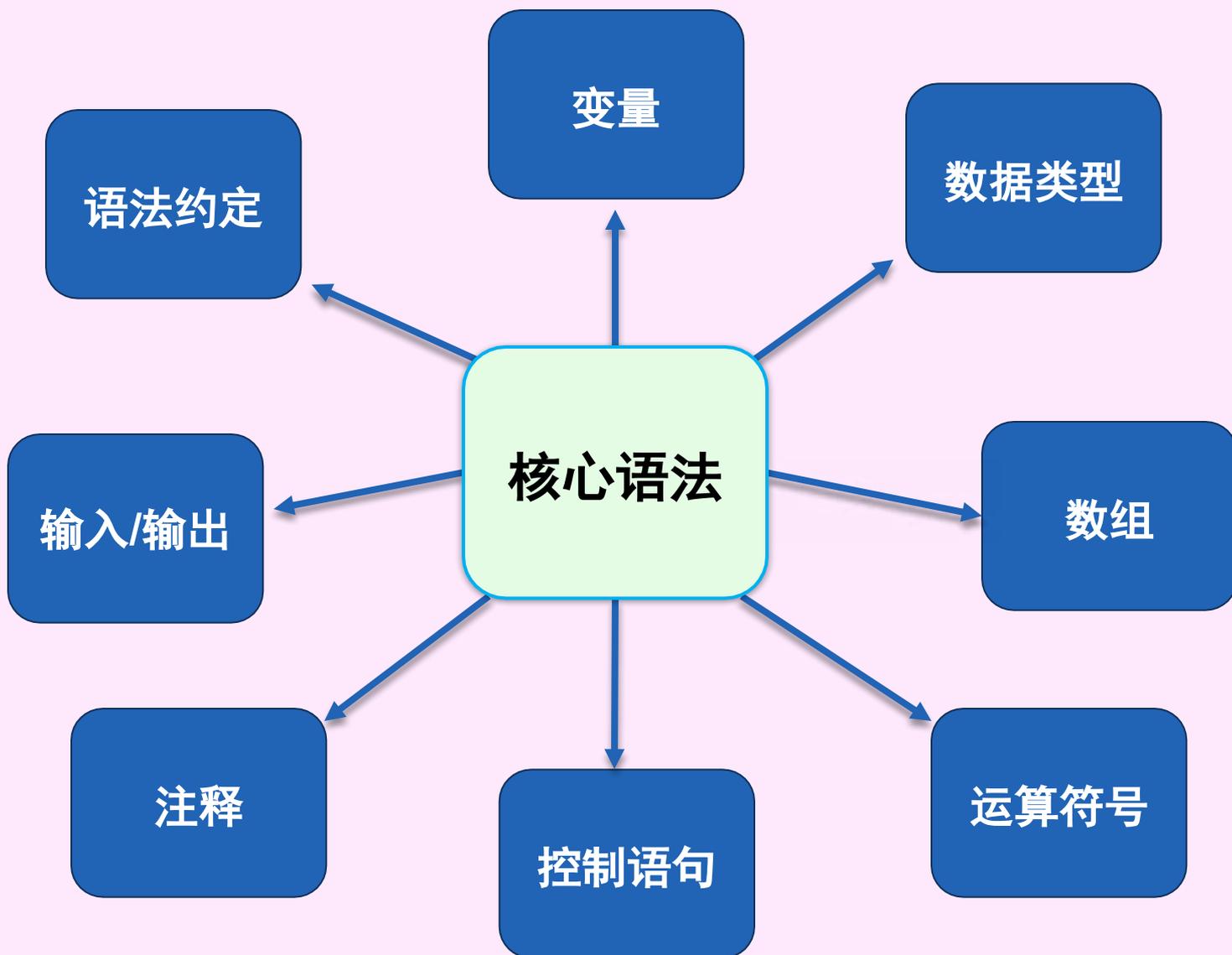
- 直接在HTML标签中

```
<input name="btn" type="button" value="弹出消息框" onclick="javascript:alert('欢迎你');"/>
```



演示示例2：网页中引用JavaScript的方式

JavaScript核心语法



■ 先声明变量再赋值

```
var width;  
width = 5;
```

var — 用于声明变量的关键字
width — 变量名

■ 同时声明和赋值变量

```
var catName= “皮皮”;  
var x, y, z = 10;
```

■ 不声明直接赋值

```
width=5;
```

 经验

变量可以不经声明而直接使用，但这种方法很容易出错，也很难查找排错，不推荐使用。

核心语法--数据类型

■ 数据类型

◆ undefined

```
var width;
```

变量width没有初始值，将被赋予值undefined

◆ null

表示一个空值，与undefined值相等

◆ number

```
var iNum=23; //整数
```

```
var iNum=23.0; //浮点数
```

◆ boolean

true和false

◆ string

一组被引号（单引号或双引号）括起来的文本

```
var string1="This is a string";
```

核心语法-- String对象

■ 属性

- ◆ 字符串对象.length

```
var str="this is JavaScript";  
var strLength=str.length; //长度是18
```

■ 方法

- ◆ 字符串对象.方法名();

| 方法名称 | 说 明 |
|---------------------------|--|
| charAt(index) | 返回在指定位置的字符 |
| indexOf(str, index) | 查找某个指定的字符串在字符串中首次出现的位置 |
| substring(index1, index2) | 返回位于指定索引index1和index2之间的字符串，并且包括索引index1对应的字符，不包括索引index2对应的字符 |
| split(str) | 将字符串分割为字符串数组 |

核心语法-- typeof运算符

- **typeof检测变量的返回值**
- **typeof运算符返回值如下：**
 - ◆ **undefined**：变量被声明后，但未被赋值
 - ◆ **string**：用单引号或双引号来声明的字符串
 - ◆ **boolean**：true或false
 - ◆ **number**：整数或浮点数
 - ◆ **object**：javascript中的对象、数组和null



演示示例3：typeof的用法

核心语法--数组2-1

■ 创建数组

表示数组的
关键字

表示数组中可存放
的元素总数

```
var 数组名称 = new Array(size);
```

■ 为数组元素赋值

```
var fruit = new Array("apple", "orange", " peach", "bananer");
```

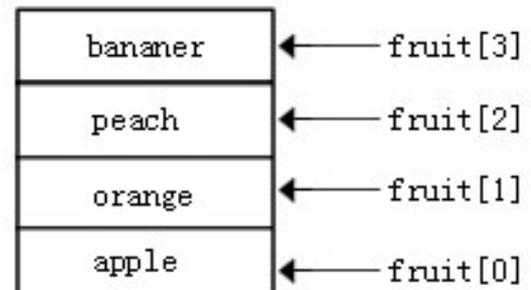
数组名[下标]

```
fruit [0] = " apple ";
```

```
fruit [1] = " orange ";
```

```
fruit [2] = " peach ";
```

```
fruit [3] = " bananer ";
```



■ 数组的常用属性和方法

| 类别 | 名称 | 描述 |
|----|---------|------------------------------|
| 属性 | length | 设置或返回数组中元素的数目 |
| 方法 | join() | 把数组的所有元素放入一个字符串，通过一个的分隔符进行分隔 |
| | sort() | 对数组排序 |
| | push() | 向数组末尾添加一个或更多 元素，并返回新的长度 |

 经验

更多方法可查阅JavaScript Array对象参考手册：



演示示例4：数组方法的应用

核心语法--运算符

| 类型 | 运算符 |
|-------|-----------------|
| 算术运算符 | + - * / % ++ -- |
| 赋值运算符 | = |
| 比较运算符 | > < >= <= == != |
| 逻辑运算符 | && ! |

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/768106132002006121>