

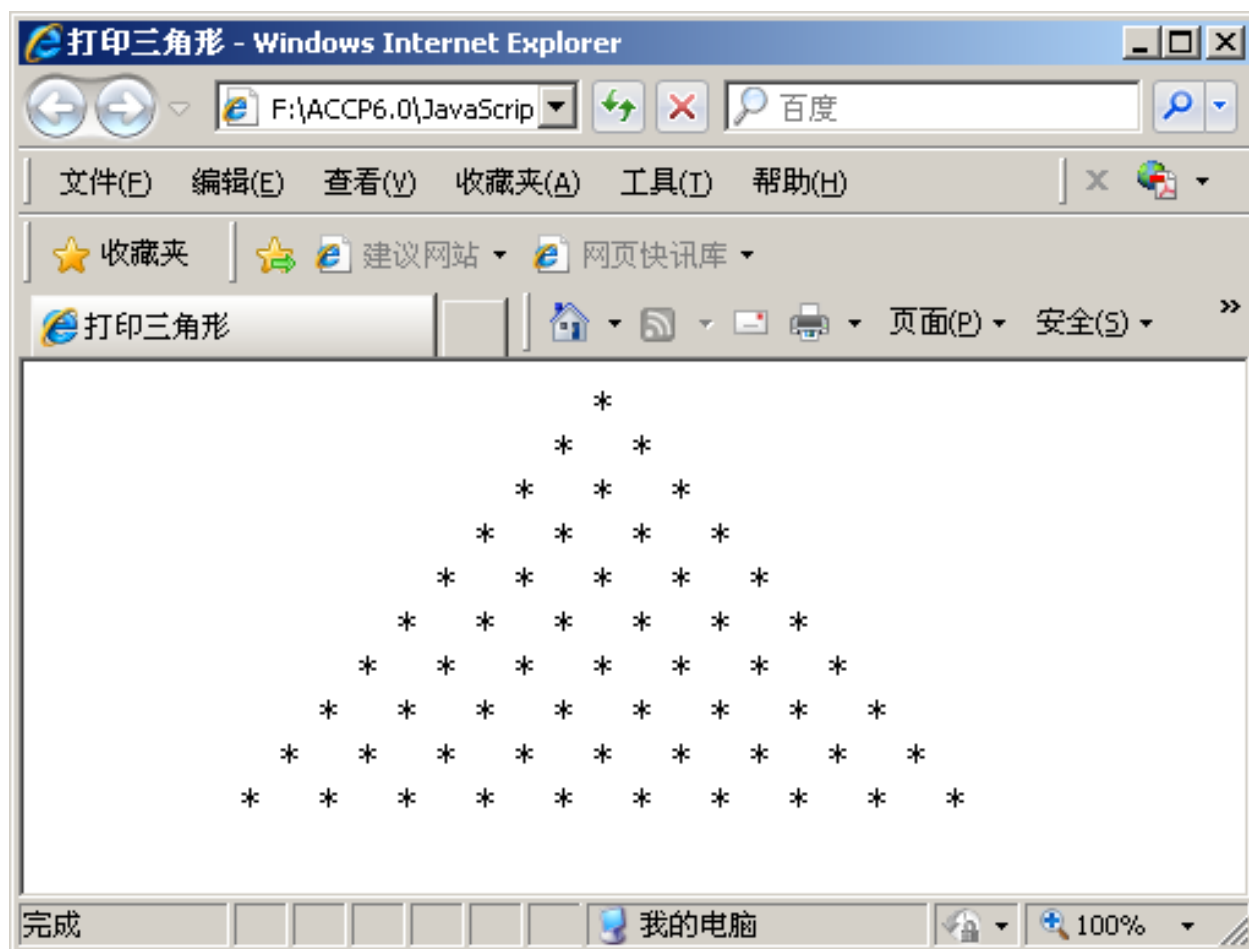
第5章 JavaScript的基本语法

本章目标

- ✓ 掌握脚本的基本结构
- ✓ 掌握JavaScript的执行原理
- ✓ 使用JavaScript实现在页面上打印三角形



本章任务





为什么要学JavaScript

- 表单验证—减轻服务器端压力

演示示例：[注册表单验证](#)

- 页面动态效果

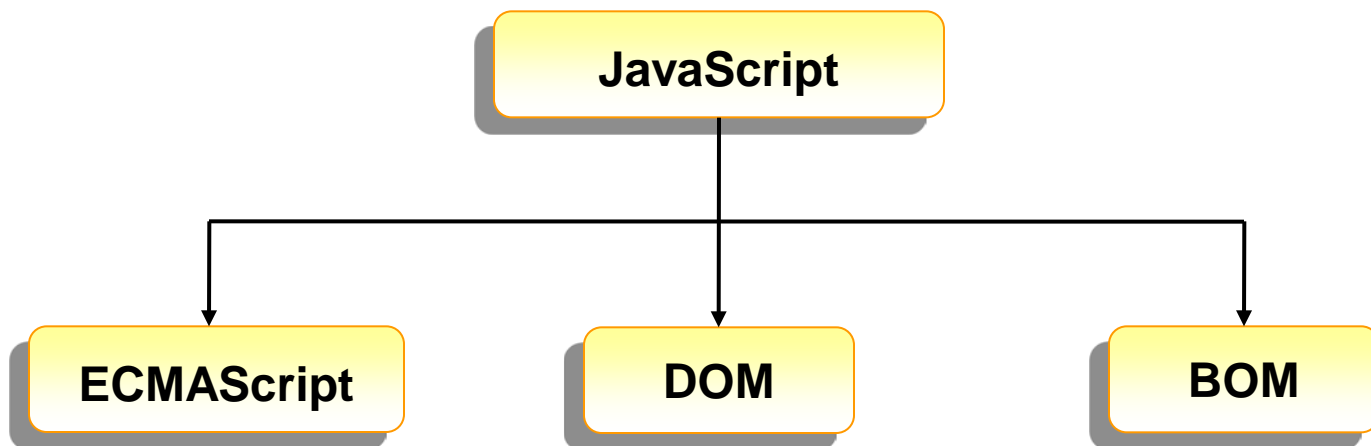
演示示例：[层的切换](#) 和 [树形菜单](#)等

- 动态改变页面内容

演示示例：[动态改变表格内容](#)

什么是JavaScript

■ JavaScript组成





ECMAScript

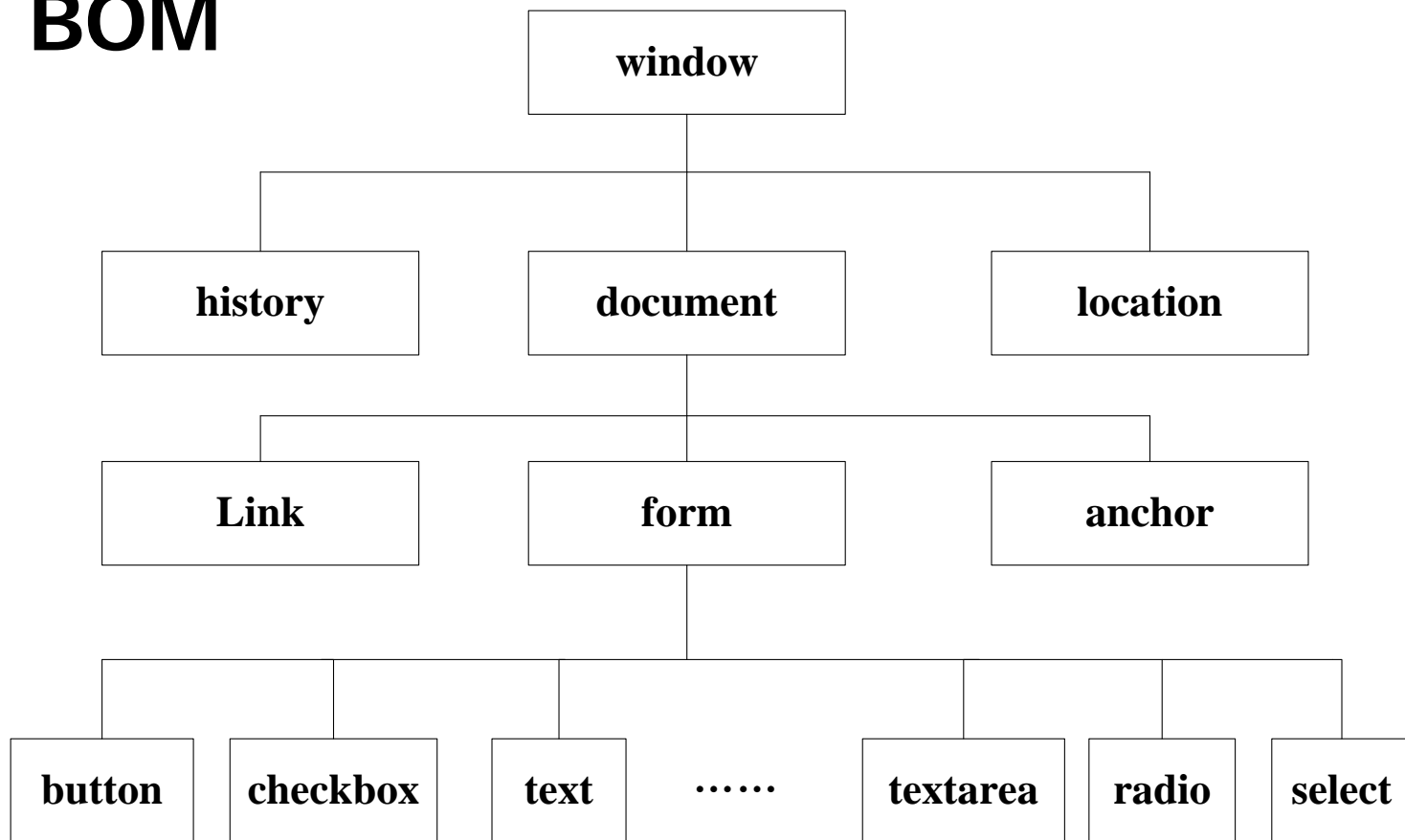
- 语法标准

- 语法
- 变量和数据类型
- 运算符
- 逻辑控制语句
- 关键字、保留字
- 对象

- 编码遵循**ECMAScript**标准

浏览器对象模型

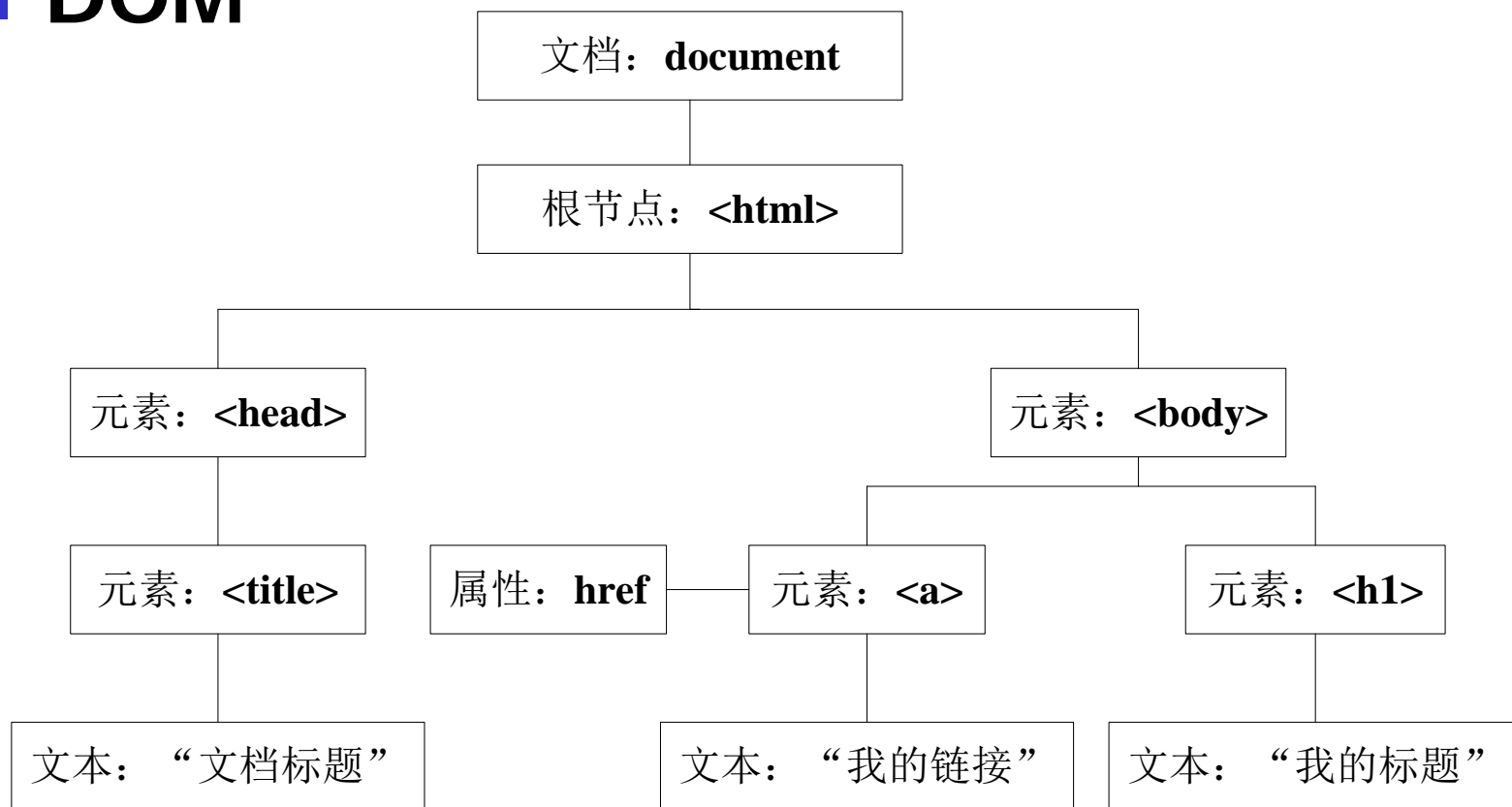
■ BOM



演示示例：弹出窗口

文档对象模型

■ DOM



演示示例: 使用DOM改变超链接

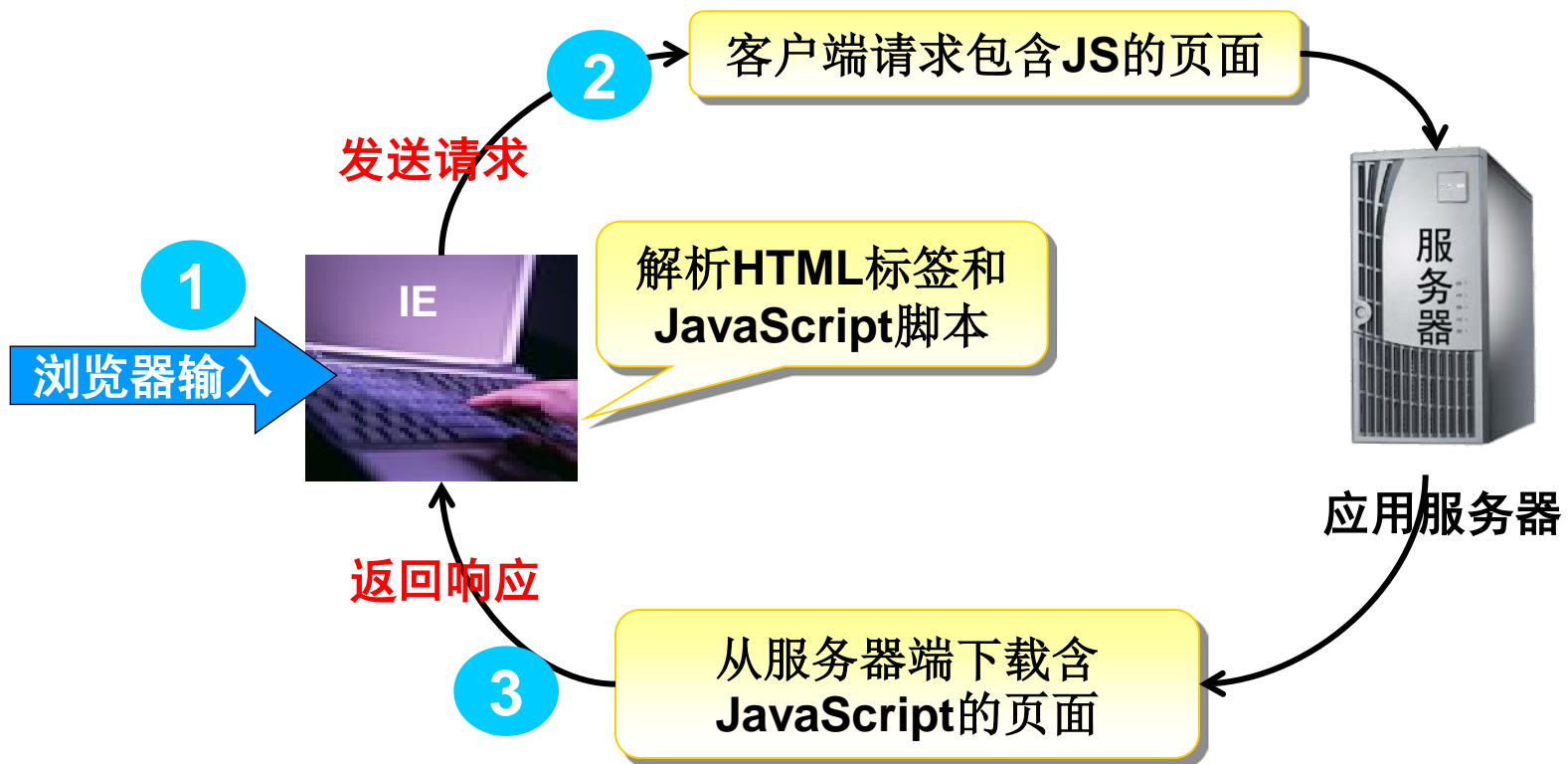


脚本的基本结构

- 脚本的基本结构

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


脚本执行原理



输出Hello World

.....

```
<title>输出Hello Word</title>
<script type="text/javascript">
document.write("使用JavaScript脚本循环输出
helloworld");
  for(var i=0;i<5;i++) {
    document.write("<h3>Hello World</h3>");
  }
  document.write("<h1>Hello World</h1>");
</script>
</head>
<body>页面主体内容</body>
</html>
```

演示示例：输出Hello World



JavaScript的使用方式

- **Html页面内嵌JS代码**
- **外部JS文件**

```
<script src="hello.js"
```

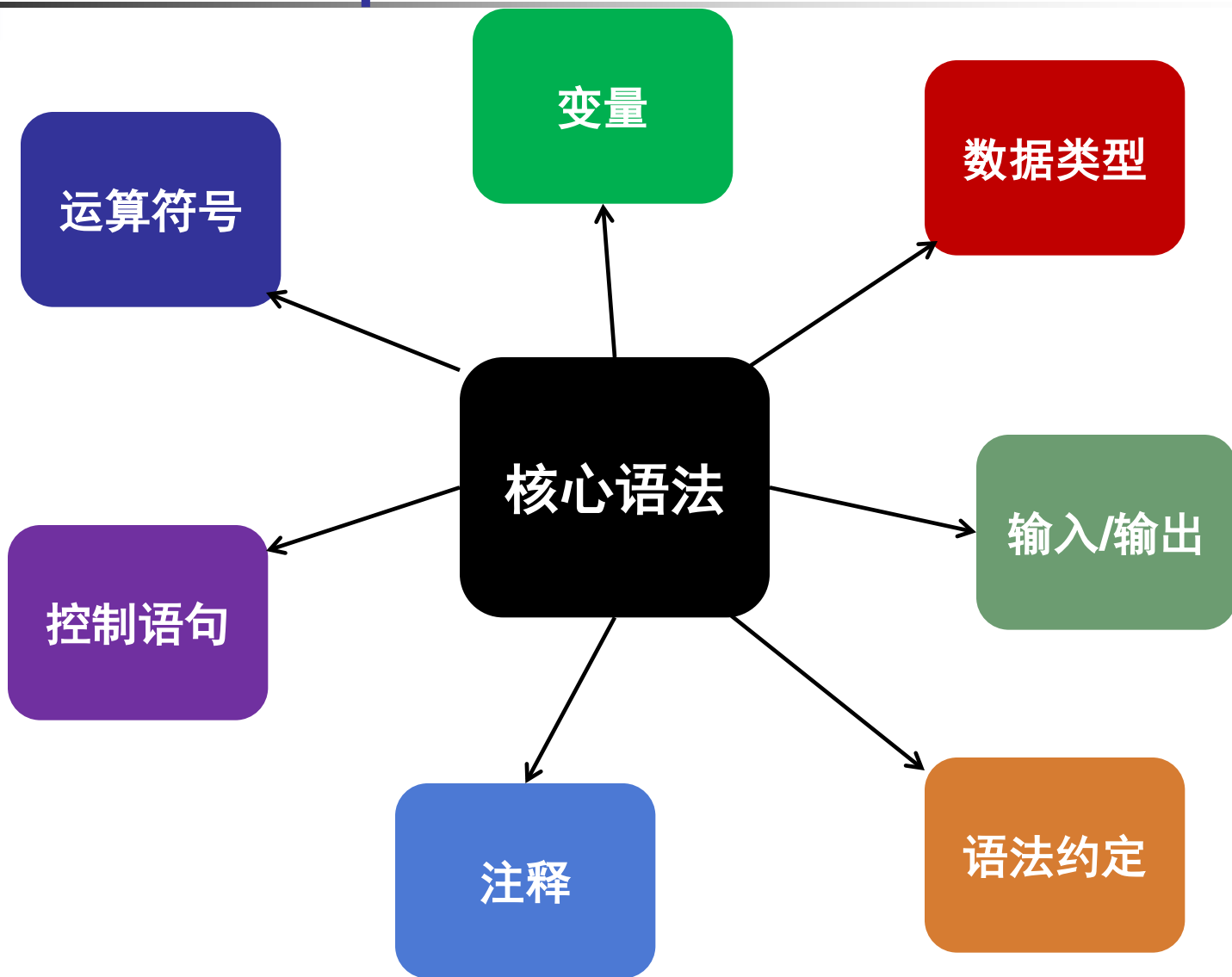
```
language="javascript"></script>
```

- **简短缩写方式**

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

演示示例: 外部JS文件 和 弹出消息框

JavaScript核心语法





变量的声明和赋值

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;
```



数据类型

- **typeof**检测变量的返回值
- 数据类型
 - **undefined**
 - **boolean**
 - **string**
 - **number**
 - **null**



typeof

- **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: [typeof的功能和用法演示](#)



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	&& !



逻辑控制语句

○ if条件语句

```
if(条件)
{
    //JavaScript代码;
}
else
{
    //JavaScript代码;
}
```



逻辑控制语句

○ switch多分支语句

```
switch (表达式)  
{    case 常量1 :  
        JavaScript语句1;  
        break;  
    case 常量2 :  
        JavaScript语句2;  
        break;  
    ...  
    default :  
        JavaScript语句3;  
}
```



逻辑控制语句

○ for、while 循环语句

```
for(初始化; 条件; 增量)
```

```
{
```

```
    JavaScript代码;
```

```
}
```

```
while(条件)
```

```
{
```

```
    JavaScript代码;
```

```
}
```

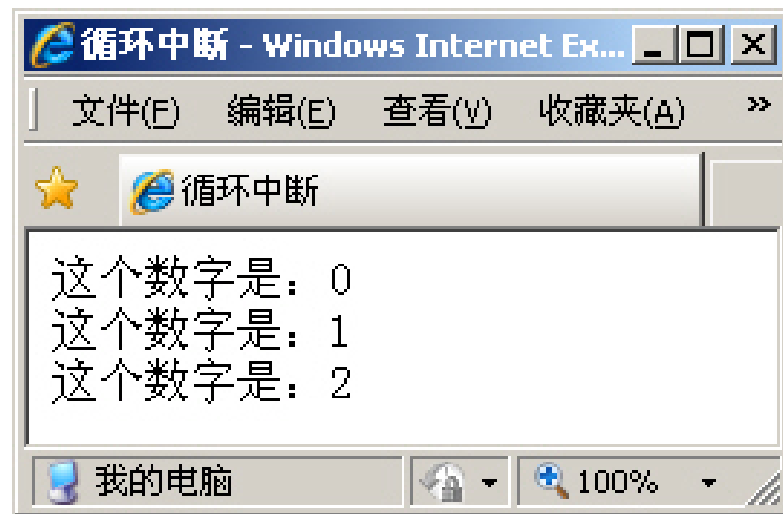
循环中断

■ break

```
<script type="text/javascript">
var i=0;
for(i=0;i<=5;i++){
    if(i==3){
        break;
    }

    document.write("这个数字是:
"+i+"<br/>");

}
</script>
```

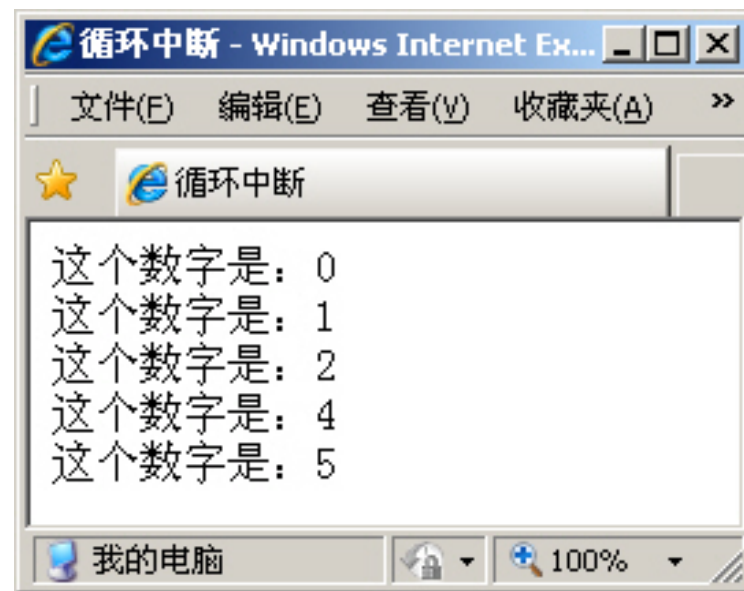


循环中断

■ continue

```
<script type="text/javascript">
var i=0;
for(i=0;i<=5;i++){
    if(i==3){
        continue;
    }

    document.write("这个数字是：
"+i+"<br/>");
}
</script>
```





注释

- 单行注释以 `//` 开始，以行末结束，例如：

```
alert("恭喜你！注册会员成功");//在页面上弹出注册会员成功的提示框
```

- 多行注释以 `/*` 开始，以 `*/` 结束，符号 `/*.....*/` 指示中间的语句是该程序中的注释。例如：

```
/*
```

```
使用for循环运行 “document.write("<h3>Hello World</h3>");” 5次  
使用document.write在页面上输出 “Hello World”
```

```
*/
```



常用的输入/输出

- **alert()**

alert("提示信息");

- **prompt()**

prompt("提示信息", "输入框的默认信息");

prompt("请输入姓名", "张三");

prompt("请输入姓名");



案例演示

- 根据输入的 次数，多次输出

```
<script type="text/javascript">
document.write("Hello World");
var j=prompt("请输入连续输出标题3的次数： ","");
for(var i=0;i<j;i++) {
    document.write("<h3>Hello World</h3>");
}
document.write("<h1>Hello World</h1>");
alert("共连续输出标题： "+j+"次");
</script>
```

演示示例： 输出HelloWorld



语法规约定

- 代码区分大小写
- 变量、对象和函数的名称
- 分号



程序调试

- **Firebug工具**

- 单步进入

- 单步跳过

- 单步退出

- **alert()方法**

演示示例: [程序调试](#)



总结



提问

- 大家简述一下JavaScript脚本的基本结构？
- JavaScript在页面中的应用有哪几种？请举例说明
- 下面使用typeof返回的值分别是什么？
 - `Typeof("rose")`
 - `Typeof(105.58)`
 - `Typeof(false)`