

Web开发（二）

--- 1-5 对象



河北师范大学软件学院
Software College of Hebei Normal University

内容提纲

- **对象简介**
- **对象的基本使用方法**
- **构造函数**
- **内置对象**
- **理解 “一切都是对象”**

内容提纲

- **对象简介**
- **对象的基本使用方法**
- **构造函数**
- **内置对象**
- **理解 “一切都是对象”**

对象简介

- 程序语言的分类：
 - 面向过程的程序语言（举例：C语言）
 - 面向对象的程序语言（举例：C++、PHP）
- JavaScript：是一门基于对象的语言
 - 具有面向对象的一部分特征
 - 在JavaScript中，一切都是对象

从现实社会谈起

- 某个具体的实物是一个对象（Object）
 - 比如：一只狗
- 对象具有一些**属性**和**行为**



在程序中实现属性和行为

- 属性：

- 通过变量来表示

- 例：

```
var name = "Lili" ;  
var breed = "Husky" ;  
var weight = 60 ;
```

- 行为

- 通过函数来实现

- 例：

```
function jump() {  
    ...  
}
```

对象的实质

- 这些描述同一个实物的数据分散存在，如何将它们统一在一起？

- JavaScript中的对象

- 一系列相关属性和方法的集合

- 属性：与对象相关的值
 - 方法：能够在对象上执行的动作

- 是一种数据类型

```
var dog = {  
  name: "Lili",  
  breed: "Husky",  
  weight: 60,  
  jump: function() {  
    //some action...  
  }  
}
```



内容提纲

- 对象简介
- 对象的基本使用方法
- 构造函数
- 内置对象
- 理解 “一切都是对象”



对象使用方法

• 定义对象

```
var objName = {  
    attrName : attrValue ,  
    methodName : function(){  
        //some code...  
    } ,  
    ...  
}
```

- 使用一对大括号表示对象，属性写在括号内
- 属性包括属性名和属性值，之间使用冒号分隔
- 属性和属性之间使用逗号分隔
- 方法同属性类似，只是值部分为一个函数体

对象使用方法

- 访问对象属性/方法

- 方式一：通过 对象名.属性名 、 对象名.方法名()
- 方式二：通过 对象名[“属性名”]

- 修改对象属性

- 直接赋值给对象属性

对象使用方法

- 遍历对象属性

- 使用for in 语句

```
for ( var i in objName ) {
```

```
    // i 在循环体内部的每次循环中代表一个属性名
```

```
    // 通过 objName[ i ] 访问到每一个属性
```

```
}
```

对象使用方法

- 添加对象属性

- 对象创建之后，要向对象添加属性，只需要为新属性赋值即可

- 删除对象属性

- 使用`delete`关键字

实例

- 天堂电影院最近播放《007》和《指环王》两部电影。使用JS程序的对象分来来存储这两部电影的数据，包括：标题、类别、评分和播放时间（每天播放3次，时间自定）。
- 思考并写出这两个对象

实例

- 增加一项功能能够显示该电影下一次播放的时间

- 方式一：定义一个函数getNextShowing

示例：1-5-6

- 方式二：扩展电影对象，增加一个方法getNextShowing

示例：1-5-7

this关键字

- this关键字在对象内部使用，指代当前对象

示例：1-5-8

内容提纲

- 对象简介
- 对象的基本使用方法
- **构造函数**
- 内置对象
- 理解 “一切都是对象”



实例中存在的问题

- 扩展方法时，每个对象都要写一个功能类似的方法
- 如果再增加一个电影，结构相同的对象需要再定义一次

代码的可重用性低

构造函数

- 能够创建出对象的一个特殊函数
 - 类似一个工厂一样，能够生产对象
 - 我们需要在工厂中设定“产品（对象）”的“模板（结构）”
 - 在“生产产品（创建对象）”时指定产品的“具体参数（属性值）”

定义电影构造函数

```
function Movie(title, showtime) {  
    this.title = title;  
    this.showtime = showtime;  
    this.rating = 0;  
    this.type = null;  
    this.getNextShowing = function() {  
        var now = "10:00";  
        for (var i=0; i<this.showtime.length ; i++)  
        {  
            if (parseInt(now) < parseInt(this.showtime[i]))  
            {  
                alert(this.title + "下一次播放时间是" + this.showtime[i]);  
                break;  
            }  
        }  
    }  
}
```



JavaScript构造函数的定义

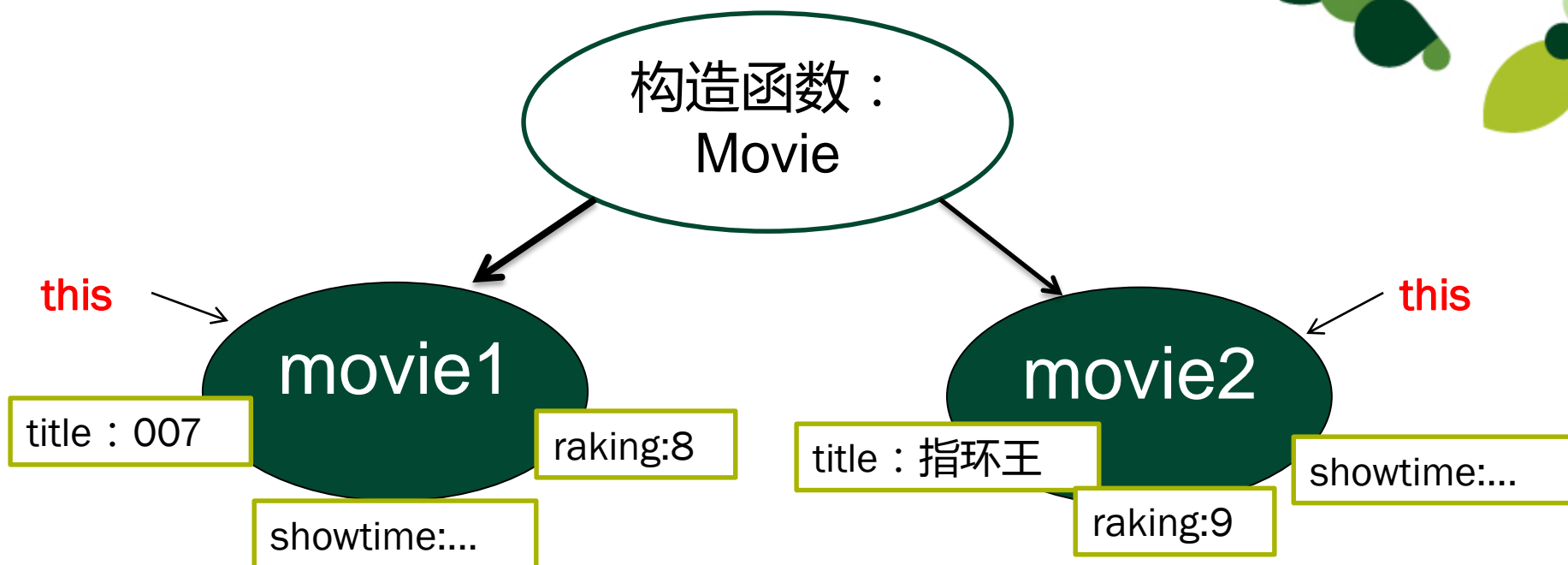
- 使用function关键字声明，为了区别普通函数，将构造函数首字母大写
- 构造函数的形参用于赋给对象的属性
- 在构造函数中，使用this指代对象
- 对象的属性和方法必须通过this访问

使用构造函数创建对象

- 构造函数不能被调用执行
- 通过new 关键字来通过构造函数创建对象
 - 将要赋给对象属性的值通过构造函数的参数传递给对象

示例：1-5-9

理解构造函数



- 构造函数相当于模板，使用new关键字创建的是对象
- 方法由哪个对象调用，方法中出现的this就指代哪个对象

内容提纲

- 对象简介
- 对象的基本使用方法
- 构造函数
- 内置对象
- 理解 “一切都是对象”



JavaScript中的对象

- JavaScript包含的对象

- 自定义对象
- 内置对象
- 浏览器对象 (BOM)
- 文档对象 (DOM)

内置对象

- JavaScript提供了多种内置对象

- String
- Array
- Math
- Date
- Event
- ...



String对象的使用

•String对象处理字符串相关操作

- 创建String对象，例：

```
var str = new String( "some string here" );
```

•常用操作

- 获取字符长度
- 判断子串位置
- 截取子串

示例：1-5-10

Array对象的使用

- Array对象处理数组相关操作

- 创建Array对象，例：

```
var movies = new Array( "007" , "指环王" );
```

- 常用操作

- 获取数组长度
- 向数组末尾增加元素
- 从数组末尾获取元素
- ...

示例：1-5-11

Math对象的使用



- Math对象包含了一系列的数学运算的功能

- Math对象不需要创建，直接使用

- 常用操作

- 四舍五入：Math.round()

- 向下取整：Math.floor()

- 生成随机数：Math.random()

- 取到最大值：Math.max()

- 取到最小值：Math.min()

示例：1-5-12

Date对象的使用

- Date对象包含了一系列的日期时间处理的功能

- 创建Date对象，例：`var now= new Date();`

- 常用操作

- 获取当前日期时间：`toLocaleString()`
 - 获取年份、月份、日期：`getFullYear()`、`getMonth()`、`getDate()`
 - 获取小时、分钟、秒钟：`getHours()`、`getMinutes()`、`getSeconds()`

参考文档

- 关于内置对象的更多使用方法，可参考：
 - http://www.w3school.com.cn/js/js_obj_date.asp
 - http://www.w3school.com.cn/js/js_reference.asp

内容提纲

- 对象简介
- 对象的基本使用方法
- 构造函数
- 内置对象
- 理解 “一切都是对象”



字符串变量和字符串对象

- 直接声明一个字符串变量和通过String对象创建的变量，在使用上有什么区别？
 - 声明一个字符串变量，该变量也具有String对象的一系列属性和方法
 - 字符串变量实质就是String对象
- 同字符串一样，数组、布尔、数值类型的变量，本质上也都是对象

回顾程序

- 那些年我们一起用过的对象：
 - `document.write()`
 - `document.getElementById("..").value`
 - ...

这些是浏览器对象和文档对象
，之后的课上将会具体学习！

内容回顾

- JavaScript的对象就是一些属性和方法的集合
- 属性的实质是一些数据，在程序中用变量保存
- 方法的实质是一些行为，在程序中用函数实现
- 创建对象，使用{ }，括号内部为属性名：属性值或方法名：方法体，属性和属性间用逗号分隔
- 通过.符号访问对象中的属性或方法
- 通过[]访问对象中的属性或方法
- 直接赋值就可以修改对象的属性值
- 为对象增加属性也通过直接赋值的方式
- 删除对象使用delete关键字



内容回顾

- 在对象内部，使用this指代当前对象
- 当相同结构的对象太多时，可以声明一个构造函数，通过构造函数创建对象可实现很好的代码可重用性
- 构造函数内部，必须使用this访问对象的属性或方法
- 构造函数不能被直接调用，使用new关键字创建对象
- JavaScript对于一些常用的数据类型及其操作封装成了内置对象：String、Array、Math、Date等
- 直接赋值的方式创建的变量，也是对象类型，具有该类对象的属性和方法，所以，“JavaScript中一切都是对象”

The background of the slide is decorated with numerous overlapping circles in various shades of green and yellow, scattered across the top and right sides.

Thank You !