

<<JavaScript设计与开发新思维>>

图书基本信息

书名：<<JavaScript设计与开发新思维>>

13位ISBN编号：9787115293411

10位ISBN编号：7115293414

出版时间：2012-10

出版单位：人民邮电出版社

作者：厄尔曼

页数：495

字数：539000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<JavaScript设计与开发新思维>>

内容概要

《JavaScript设计与开发新思维》是JavaScript最新技术的权威图书。

书中不仅包括JavaScript的基本概念，还详细叙述了其与HTML之间的关系。

《JavaScript设计与开发新思维》将向读者介绍变量、常用运算符和控制结构的内容，以及如何创建函数、处理事件和更多HTML表单相关的操作。

这些内容不仅能帮助读者使用JavaScript，还能帮助读者透彻地了解这种语言。

阅读完本书，读者将能掌握Ajax技术、框架知识，并能使用JavaScript和PHP语言创建一个完整的网页。

《JavaScript设计与开发新思维》适合JavaScript初学者或者是其他网页开发爱好者学习使用。

作者简介

Larry

Ullman是一位作家、Web和软件开发人员、培训师、教师、演说家和顾问。

他已经著有22本书籍和数十篇论文。

Larry的读者可以证明，他的长处是将难以理解的行话转换为自然语言：将技术性而晦涩难懂的知识转化为容易理解和实用的内容。

书籍目录

第1部分 入门

第1章 (重新)介绍JavaScript

- 1.1 什么是JavaScript
- 1.2 JavaScript的历史
 - 1.2.1 间或有些丑陋的历史
 - 1.2.2 Ajax重临
 - 1.2.3 浏览器改进
 - 1.2.4 框架的兴起
- 1.3 JavaScript不是什么
- 1.4 JavaScript和其他常见技术的比较
- 1.5 为什么说JavaScript是一种好语言
- 1.6 JavaScript版本和浏览器支持
- 1.7 JavaScript编程目标
- 1.8 小结

第2章 JavaScript实战

- 2.1 选择DOCTYPE
- 2.2 HTML5入门
 - 2.2.1 一个HTML5模板
 - 2.2.2 HTML5表单元素
- 2.3 为HTML添加JavaScript
- 2.4 关键的开发方法
 - 2.4.1 功能退化
 - 2.4.2 渐进增强
 - 2.4.3 无干扰的JavaScript
- 2.5 拼凑一些代码
 - 2.5.1 开发基本功能
 - 2.5.2 添加JavaScript层次
 - 2.5.3 处理事件
 - 2.5.4 执行验证
- 2.6 窃取这段JavaScript
- 2.7 小结

第3章 行业工具

- 3.1 大辩论：文本编辑器还是IDE？
 - 3.1.1 共有特性
 - 3.1.2 两者的比较
 - 3.1.3 一些文本编辑器
 - 3.1.4 几种IDE
- 3.2 浏览器：你的朋友，你的敌人
 - 3.2.1 Google公司的Chrome
 - 3.2.2 Mozilla公司的Firefox
 - 3.2.3 微软公司的Internet Explorer
 - 3.2.4 Opera
 - 3.2.5 苹果公司的Safari
- 3.3 在多个浏览器上测试

<<JavaScript设计与开发新思维>>

- 3.4 测试JavaScript
- 3.5 错误和调试
 - 3.5.1 错误类型
 - 3.5.2 常见错误起因
 - 3.5.3 调试技术
 - 3.5.4 使用Firebug
- 3.6 在线资源
- 3.7 小结
- 第2部分 JavaScript基础
- 第4章 简单变量类型
 - 4.1 变量基础知识
 - 4.1.1 变量声明
 - 4.1.2 变量名
 - 4.1.3 赋值
 - 4.1.4 简单值类型
 - 4.2 使用数字
 - 4.2.1 算术运算符
 - 4.2.2 创建计算器
 - 4.2.3 格式化数字
 - 4.2.4 Math对象
 - 4.3 使用字符串
 - 4.3.1 创建字符串
 - 4.3.2 字符串拆分
 - 4.3.3 操纵字符串
 - 4.3.4 转义序列
 - 4.4 完成类型转换
 - 4.5 复习与提高
 - 4.5.1 复习
 - 4.5.2 提高
 - 4.6 小结
- 第5章 使用控制结构
 - 5.1 条件语句的基础知识
 - 5.1.1 if条件语句
 - 5.1.2 什么是真
 - 5.1.3 比较运算符
 - 5.1.4 逻辑运算符
 - 5.1.5 汇总
 - 5.2 更多的条件语句
 - 5.2.1 if-else条件语句
 - 5.2.2 if-else if条件语句
 - 5.2.3 switch条件语句
 - 5.2.4 神秘的条件语句
 - 5.3 更复杂的条件
 - 5.3.1 数值比较
 - 5.3.2 字符串比较
 - 5.3.3 typeof运算符
 - 5.4 循环基础知识

- 5.4.1 for循环
- 5.4.2 while循环
- 5.5 复习和提高
 - 5.5.1 复习
 - 5.5.2 提高
- 5.6 小结
- 第6章 复杂变量类型
 - 6.1 生成日期和时间
 - 6.1.1 创建日期
 - 6.1.2 日期方法
 - 6.1.3 使用时区
 - 6.1.4 更改日期
 - 6.1.5 日期运算
 - 6.2 使用数组
 - 6.2.1 创建数组
 - 6.2.2 访问数组元素
 - 6.2.3 访问所有数组元素
 - 6.2.4 删除数组元素
 - 6.2.5 数组方法
 - 6.2.6 在字符串和数组之间转换
 - 6.3 使用对象
 - 6.3.1 创建对象
 - 6.3.2 访问对象属性
 - 6.3.3 访问所有对象属性
 - 6.3.4 删除对象属性
 - 6.4 数组与对象的比较
 - 6.5 复习和提高
 - 6.5.1 复习
 - 6.5.2 提高
 - 6.6 小结
- 第7章 创建函数
 - 7.1 基础知识
 - 7.1.1 定义你自己的函数
 - 7.1.2 向函数传递值
 - 7.1.3 验证函数参数
 - 7.1.4 参数值传递方式
 - 7.1.5 从函数返回值
 - 7.1.6 理解变量作用域
 - 7.2 作为对象的函数
 - 7.2.1 作为变量值的函数
 - 7.2.2 作为参数值的函数
 - 7.2.3 汇总
 - 7.3 更有趣的内容
 - 7.3.1 上下文和this对象
 - 7.3.2 匿名函数
 - 7.3.3 嵌套函数
 - 7.3.4 执行递归

7.4 复习和提高

7.4.1 复习

7.4.2 提高

7.5 小结

第8章 事件处理

8.1 事件处理的基础

8.2 创建事件监听器

8.2.1 (不要使用)嵌入式事件处理器

8.2.2 传统事件处理

8.2.3 W3C事件处理

8.2.4 IE浏览器事件处理

8.2.5 创建一个事件分配器

8.3 创建一个实用程序库

8.4 事件类型

8.4.1 输入设备事件

8.4.2 键盘事件

8.4.3 浏览器事件

8.4.4 表单事件

8.5 事件可访问性

8.6 事件和渐进增强

8.7 高级事件处理

8.7.1 引用事件

8.7.2 事件属性

8.7.3 检查按键

8.7.4 阻止默认事件行为

8.7.5 事件的两个阶段

8.7.6 委派事件处理

8.8 复习和提高

8.8.1 复习

8.8.2 提高

8.9 小结

第9章 JavaScript与浏览器

9.1 使用对话框

9.1.1 警告对话框

9.1.2 确认对话框

9.1.3 提示对话框

9.1.4 自定义对话框

9.2 处理窗口

9.2.1 window全局对象

9.2.2 操纵窗口的大小和位置

9.2.3 屏幕属性

9.2.4 创建新窗口

9.2.5 自定义弹出窗口

9.2.6 改变焦点

9.2.7 可访问解决方案

9.2.8 窗口之间的通信

9.2.9 处理浏览器历史

- 9.2.10 重定向浏览器
- 9.2.11 创建具有表现力的URL
- 9.2.12 打印本页
- 9.2.13 文档对象
- 9.3 操纵DOM
 - 9.3.1 DOM基础
 - 9.3.2 DOM快捷方式
 - 9.3.3 使用DOM操纵
 - 9.3.4 DOM方法
 - 9.3.5 CSS选择符
 - 9.3.6 修改元素
 - 9.3.7 创建元素
- 9.4 JavaScript与CSS
 - 9.4.1 引用单独的样式
 - 9.4.2 隐藏和显示元素
 - 9.4.3 创建模态窗口
 - 9.4.4 引用样式表
- 9.5 使用Cookie
 - 9.5.1 创建Cookie
 - 9.5.2 读取Cookie
 - 9.5.3 删除Cookie
 - 9.5.4 创建一个Cookie程序库
 - 9.5.5 使用Cookie程序库
- 9.6 使用定时器
- 9.7 复习与提高
 - 9.7.1 复习
 - 9.7.2 提高
- 9.8 小结
- 第10章 处理表单
 - 10.1 表单一般考虑事项
 - 10.1.1 创建错误信息
 - 10.1.2 创建工具提示
 - 10.1.3 禁用提交按钮
 - 10.2 文本输入域和文本区域
 - 10.2.1 文本输入域和文本区域基础知识
 - 10.2.2 实现自动完成
 - 10.3 选择菜单
 - 10.3.1 验证选择菜单
 - 10.3.2 创建动态选择框
 - 10.4 复选框
 - 10.5 单选按钮
 - 10.6 处理文件上传
 - 10.7 正则表达式
 - 10.7.1 什么是正则表达式
 - 10.7.2 创建一个正则表达式
 - 10.7.3 正则表达式函数
 - 10.7.4 定义简单的模式

<<JavaScript设计与开发新思维>>

- 10.7.5 使用限定符
- 10.7.6 使用字符类
- 10.8 汇总
- 10.9 复习与提高
 - 10.9.1 复习
 - 10.9.2 提高
- 10.10 小结
- 第11章 Ajax
 - 11.1 Ajax基础知识
 - 11.1.1 理解Ajax
 - 11.1.2 创建一个Ajax对象
 - 11.1.3 指定结果处理器
 - 11.1.4 发出请求
 - 11.1.5 处理服务器响应
 - 11.1.6 发送数据
 - 11.1.7 基本调试
 - 11.2 处理其他数据
 - 11.2.1 XML
 - 11.2.2 JSON
 - 11.3 服务器端脚本
 - 11.3.1 返回普通文本
 - 11.3.2 返回XML
 - 11.3.3 返回JSON
 - 11.4 Ajax示例
 - 11.4.1 显示进度
 - 11.4.2 完成第2章中的登录示例
 - 11.4.3 创建一个Ajax联络表单
 - 11.4.4 预加载数据
 - 11.4.5 使用定时器的股票报价
 - 11.5 复习与提高
 - 11.5.1 复习
 - 11.5.2 提高
 - 11.6 小结
- 第3部分 后续步骤
- 第12章 错误管理
 - 12.1 捕捉和抛出错误
 - 12.1.1 捕捉错误
 - 12.1.2 finally子句
 - 12.1.3 抛出异常
 - 12.1.4 汇总
 - 12.2 使用断言
 - 12.2.1 创建断言函数
 - 12.2.2 Firebug中的断言
 - 12.3 单元测试
 - 12.3.1 安装jsUnity
 - 12.3.2 定义测试
 - 12.3.3 运行测试

- 12.3.4 记录结果
- 12.3.5 建立和拆除
- 12.3.6 汇总
- 12.4 复习与提高
 - 12.4.1 复习
 - 12.4.2 提高
- 12.5 小结
- 第13章 框架
 - 13.1 选择框架
 - 13.2 jQuery简介
 - 13.2.1 jQuery入门
 - 13.2.2 选择元素
 - 13.2.3 操纵元素
 - 13.2.4 DOM操纵
 - 13.2.5 处理事件
 - 13.2.6 创建特效
 - 13.2.7 实施Ajax操作
 - 13.2.8 jQuery插件
 - 13.3 YUI简介
 - 13.3.1 YUI入门
 - 13.3.2 选择元素
 - 13.3.3 操纵元素
 - 13.3.4 DOM操纵
 - 13.3.5 处理事件
 - 13.3.6 创建特效
 - 13.3.7 执行Ajax
 - 13.3.8 使用小部件和工具
 - 13.4 库
 - 13.5 复习与提高
 - 13.5.1 复习
 - 13.5.2 提高
 - 13.6 小结
- 第14章 高级JavaScript
 - 14.1 定义命名空间
 - 14.2 创建自定义对象
 - 14.2.1 创建单个自定义对象
 - 14.2.2 创建自定义对象的多个实例
 - 14.2.3 更完整的自定义对象
 - 14.2.4 汇总
 - 14.3 理解原型
 - 14.3.1 原型化继承
 - 14.3.2 添加原型方法
 - 14.4 使用闭包
 - 14.5 备选类型标识
 - 14.6 精简代码
 - 14.7 复习与提高
 - 14.7.1 复习

14.7.2 提高

14.8 小结

第15章 PHP和JavaScript的结合

15.1 确定目标

15.2 创建数据库

15.3 建立网站

15.4 编写非JavaScript版本

15.4.1 列出拍卖

15.4.2 登录

15.4.3 查看单个拍卖

15.5 创建Ajax资源

15.5.1 登录Ajax脚本

15.5.2 用于出价的Ajax脚本

15.5.3 获取出价的Ajax脚本

15.6 添加JavaScript

15.6.1 编写login.js

15.6.2 在View.php中编写JavaScript

15.6.3 编写view.js

15.7 完成示例

15.8 复习与提高

15.8.1 复习

15.8.2 提高

15.9 小结

章节摘录

版权页：插图：使用保留字 JavaScript有二十多个保留字，如var、function等。

你不能使用这些保留字作为变量或者函数名称。

尽管如此，我从不倾向于在书里包含保留字的列表：许多在线资源已经为你做到了这一点，列表非常冗长，以至于无法记住。

但是，如果你为所创建的变量和函数使用描述性的名称，就不太可能与较为通用的保留字冲突。

3.5.3调试技术 理解了基本的错误类型和常见起因，我们再来了解一些调试技术。

选择好的文本编辑器或者IDE 在这里不作过多的研究，但是选择和掌握一种好的文本编辑器或者IDE将使你的JavaScript编程工作变得更加简单。

毕竟，这是它存在的理由！

选择好的开发浏览器 这个主题在本章前面也已经讨论过：选择好的浏览器和合适的扩展（在合适的时候），并且学习最大限度地发挥其作用的方法。

在任何时候都打开浏览器的控制台 不管结果如何，浏览器在出现错误（意味着可能有你未意识到的问题）时不会对此大做文章。

保持浏览器错误控制台可见，你就能看到发生的问题。

使用JavaScript验证器 正如HTML验证服务一样，JavaScript也有验证服务。

JavaScript大师Douglas Crockford创建的JSLint（www.jshint.com）就是这样的网站。

JSLint是“代码质量工具”，能够识别有疑问以及潜在问题的代码。

从JSLint派生出来的JSHint（www.jshint.com）是一个更加令人愉快的替代产品。

对使用JSLint的争论是缘于JSLint相当保守和严格，倡导的是Crockford认为你应该遵循的风格。

JSHint的目的相同，但是可以对代码质量问题进行灵活的定制。

<<JavaScript设计与开发新思维>>

编辑推荐

简单易懂的渐进式指导，丰富的插图和清晰的示例突出讲解实用的开发技术，专注于提高实战能力蕴含作者最为资深Web专家多年积累的宝贵经验强调创建可靠地工作于现代浏览器和设备上（甚至是不支持JavaScript的）的代码策略

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>