

JS入门

数据类型

- 数据类型
 - null
 - undefined
 - number
 - string
 - boolean
 - object
- 判断数据类型
 - typeof
 - instanceof
 - Object.prototype.toString

运算符

- 算术运算符
 - 加法运算符
 - 余数运算符
 - 自增自减运算符
 - 数值运算符 (+)
 - 指数运算符
- 比较运算符
 - 字符串的比较
 - 严格相等运算符
 - 相等运算符
- 布尔运算符
 - 取反运算符 (!)
 - 且运算符 (&&) "短路"机制
 - 或运算符 (||)
 - 三元运算符 (?:)
- 二进制位运算符
 - 或运算符 (|)
 - 与运算符 (&)
 - 取反运算符 (~)
 - 异或运算符 (^)
 - 左移运算符 (<<)
 - 右移运算符 (>>)
 - 头部补零的右移运算符 (>>>)
- 其它运算符
 - void运算符
 - 逗号运算符
 - 大部分左结合运算
 - 右结合运算, 赋值和三元运算符

语法专题

- 错误处理机制
 - error实例
 - SyntaxError (解析错误)
 - ReferenceError (引用不存在的变量)
 - RangeError (超出有效范围)
 - TypeError (类型错误)
 - URIError (URI函数参数错误)
 - EvalError (eval执行错误)
 - try...catch
 - finally (不管是否错误都会执行)
- 编程风格
 - 缩进
 - 区块
 - 相等与严格相等
 - 语句的合并
 - 自增自减运算符
- console对象与控制台
 - log、info、debug
 - warn、error
 - table、count
 - group、groupEnd、groupCollapsed
 - debugger

标准库

- Object对象
- 属性描述对象
- Array对象
- 包装对象
- Boolean对象
- Number对象
- String对象
- Math对象
- RegExp对象
- JSON对象

面向对象编程

- 实例对象与new命令
 - 构造函数
 - new的原理
 - new.target (指向当前函数)
 - Object.create()
- this关键字
 - 含义: 属性或方法"当前"所在对象
 - 使用场合
 - 避免多层this、避免回调中的this
 - 绑定this的方法: call、apply、bind
- 对象的继承
 - 原型链的方式
 - 盗用构造函数
 - 组合继承
 - 原型式继承
 - 寄生式组合继承
- Object对象的相关方法
 - getPrototypeOf、setPrototypeOf
 - create
 - getOwnPropertyNames
 - hasOwnProperty
 - in和for...in
- 严格模式
 - 显式报错
 - 只读属性不可写
 - 只设置了取值器的属性不可写
 - 禁止扩展的对象不可扩展
 - eval、arguments不可用作标识名
 - 函数不能有重名的参数
 - 增强的安全措施
 - 全局变量必须显式声明
 - 禁止this指向全局对象
 - 禁止使用callee、caller
 - 禁止删除变量
 - 静态绑定
 - 禁止使用with
 - eval作用域
 - 向下个版本的JS过渡
 - 非函数代码块不能声明函数
 - 新增一系列保留字

异步操作

DOM

事件

浏览器模型

网页元素接口