

面向对象

课堂主题

面向对象(三)

课堂目标

1. 基于观察者模式的事件机制
2. 掌握原型链继承的实现方式
3. 掌握深拷贝和浅拷贝

上节课重点内容回顾

ES6 继承

正课内容

实现基于观察者模式实现 **Event** 类

当对象间存在一对多关系时，则使用观察者模式（Observer Pattern）。比如，当一个对象被修改时，则会自动通知它的依赖对象。观察者模式属于行为型模式。

- 事件池
- 监听器
- 解除监听
- 触发器

class 的其他项

- static 静态属性(方法)

class 注意事项

- 命名问题
- 严格模式
- 预解析问题

JavaScript 严格模式 - 扩展

- "use strict"
- 严格模式有点 消除JavaScript语法的一些不合理、不严谨之处，减少一些怪异行为; - 消除代码运行的一些不安全之处，保证代码运行的安全; - 提高编译器效率，增加运行速度; - 为未来新版本的JavaScript做好铺垫。"严格模式"体现了JavaScript更合理、更安全、更严谨的发展方向
- 严格模式先的限制
 - 不允许使用未声明的变量;
 - 不允许变量重名;
 - 不允许使用八进制;

- 不允许对只读属性赋值;
- 不允许使用转义字符;
- 变量名不能使用 "eval" 字符串;
- 变量名不能使用 "arguments" 字符串;
- 禁止this关键字指向全局对象; https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Strict_mode

ES5 继承

构造函数继承

原型继承

拷贝继承

深拷贝 和 浅拷贝

总结

- 深拷贝和浅拷贝的区别
- 原型继承
- 观察者模式

下节课内容

- JQ 核心源码探秘