

千锋HTML5学院

# 第二阶段javascript课程课件



1

对象Object

2

对象的创建

3

日期Date

4

定时器

# 对象

什么是对象，其实就是一种类型，即引用类型。而对象的值就是引用类型的实例。在 ECMAScript 中引用类型是一种数据结构，用于将数据和功能组织在一起。

什么是类，在传统的面向对象的语言中，类是对象的抽象，对象是类的具体。由类来创建对象。但 ECMAScript 中却没有这种东西。虽然 ECMAScript 是一门面向对象的语言，却不具备传统面向对象语言所支持的类的基本结构。

# 对象的创建

## Object 类型

创建 Object 类型有两种。

一种是使用 new 运算符，另一种是字面量表示法。

### 1. 使用 new 运算符创建 Object

```
var box = new Object();           //new方式  
box.name = '张三';               //创建属性字段  
box.age = 18;                    //创建属性字段
```

new 关键字可以省略

```
var box = Object();              //省略了new关键字, 不建议
```

### 2. 使用字面量方式创建 Object

```
var box = {                      //字面量方式  
    name : '张三',              //创建属性字段, 最后加逗号  
    age : 18  
};
```

## 对象的创建

### 3. 属性字段也可以使用字符串形式

```
var box={  
    'name': '张三', //也可以用字符串形式  
    'age': 28  
};
```

### 4. 使用字面量及传统赋值方式

```
var box={};           //字面量方式声明空的对象  
box.name='张三';      //点符号给属性赋值  
box.age= 18;
```

### 5. 两种属性输出方式

```
alert(box.age);        //点表示法输出  
alert(box['age']);     //中括号表示法输出，注意引号
```

## 给对象的创建方法

### 6. 给对象创建方法

```
var box={  
    //对象中添加方法(函数)run  
    run:function() {  
        retrun "正在跑步.." ;  
    }  
}  
alert(box.run()); //调用对象中的方法
```

### 7. 使用 delete 删除对象属性

```
delete box.name; //删除属性
```

## 练习

- 1, 创建一个人的对象, 添加属性: 姓名 年龄 爱好 薪资期望, 并有一个打印自身信息的方法, 可以输出自身信息;
- 2, 创建一个锤子对象, 有属性: 宽, 高, 重; 方法: 可以锤钉子
- 3, 有一辆50km/h车, 跑在一条1000km路上, 问多少小时跑完?  
对象:
  - 车Car 属性: 速度speed 50km/h  
功能: 跑在路上runOnRoad(Road)
  - 路Road 属性: 长度length 1000km



## 日期对象Date

### Date对象代表日期

#### Date

类型使用自 UTC（Coordinated Universal Time，国际协调时间）1970 年 1 月 1 日午夜（零时）开始经过的毫秒数来保存日期。**Date** 类型保存的日期能够精确到 1970 年 1 月 1 日之前或之后的 285616 年。

创建一个日期对象，使用 **new** 运算符和 **Date** 构造方法(构造函数)即可。

```
var d = new Date();
```

在调用**Date**构造方法而不传递参数的情况下，新建的对象自动获取当前的时间和日期。

创建日期对象并指定时间

```
var d = new Date("2015/08/22");
```

```
var d = new Date(2016,04,13,14,34);
```

【注】日期的格式可以是 “2015/08/22” ， “2015-08-22” ， 或1970 年当前日期的毫秒数56521211021



## 日期对象Date(格式化方法)

<b>alert(box.toString());</b>	以特定的格式显示星期几、月、日和年
<b>alert(box.getTimeString());</b>	以特定的格式显示时、分、秒和时区
<b>alert(box.toLocaleDateString());</b>	以特定地区格式显示年、月、日
<b>alert(box.toLocaleTimeString());</b>	以特定地区格式显示时、分、秒
<b>alert(box.toUTCString());</b>	以特定的格式显示完整的 UTC 日期: 年,月,日,时,分,秒。



## 日期对象Date

```
Date.parse("2015-08-24");
```

//转换格式默认支持2015-08-24或2015/08/24

//返回距离1970年1月1日0时的毫秒数

```
d.getTime() / d.setTime()
```

getTime 获取某个日期自1970年以来的毫秒数

setTime 修改日期的毫秒数，对应的日期时间也会修改

```
var myDate = new Date();  
myDate.getTime();
```



## 日期对象Date

- set / getDate()** 从Date对象中返回一个月中的某一天(1~31)
- getDay()** 从Date对象返回一周中的某一天(0~6)
- set / getMonth()** 从Date对象中返回月份(0~11)
- set / getFullYear()** 从Date对象以四位数返回年份
- set / getHours()** 返回Date对象的小时(0~23)
- set / getMinutes()** 返回Date对象的分钟(0~59)
- set / getSeconds()** 返回Date对象的秒数(0~59)
- set / getMilliseconds()** 返回Date对象的毫秒
- set / getTime()** 返回1970年1月1日至今的毫秒数
- getTimezoneOffset()** 返回本地时间与格林尼治标准时间(GMT)的分钟差

## 日期对象Date

- set / getUTCDate() 根据世界时从Date对象返回月中的一天(1~31)
- getUTCDay() 根据世界时从Date对象返回周中的一天(0~6)
- set / getUTCMonth() 根据世界时从Date对象返回月份(0~11)
- set / getUTCFullYear() 根据世界时从Date对象返回四位年份
- set / getUTCHours() 根据世界时返回Date对象的小时(0~23)
- set / getUTCMinutes() 根据世界时返回Date对象的分钟(0~59)
- set / getUTCSeconds() 根据世界时返回Date对象的秒数(0~59)
- set / getUTCMilliseconds() 根据世界时返回Date对象的毫秒(0~999)

## 日期对象Date

- `toString()` 把**Date**对象转换为字符串
- `getTimeString()` 把**Date**对象的时间部分转换为字符串
- `toDateString()` 把**Date**对象的日期部分转换成字符串
- `toUTCString()` 根据世界时,把**Date**对象转换为字符串
- `toLocaleString()` 根据本地时间格式,把**Date**转换为字符串
- `toLocaleTimeString()` 根据本地时间格式, 把**Date**时间部分转换为字符串
- `toLocaleDateString()` 根据本地时间格式,把**Date**对象的日期部分转换为字符串
- `UTC()` 根据世界时返回1970年1月1日到指定日期的毫秒数
- `valueOf()` 返回**Date**对象的原始值



## 日期对象Date

注意：

- 月份是从0-11，所以取得的值要加1才是当前月份
- 星期天是从0-6, 0代表星期天



## 示例

1、显示当前时间 (年 月 日 时 分 秒)

2、`setDate()`和`getDate()`，封装一个函数，包含两个参数日期`date`和天数`n`,可以根据输入的`n`(天数)显示日期`date`在`n`天后的时间

提示：可以使用`while`循环, 将`n`(天数)递减, 直到`n+day<本月的总天数`



## 定时器: setInterval ( ) 函数

`setInterval()` 方法可按照指定的周期（以毫秒计）来调用函数或计算表达式

`setInterval()` 方法会不停地调用函数，直到 `clearInterval()` 被调用或窗口被关闭。由 `setInterval()` 返回的 ID 值可用作 `clearInterval()` 方法的参数。

### 创建定时器

#### `setInterval(code,millisec)`

`code` 是要调用的代码块或者函数

`millisec` 是周期性执行代码块或函数的间隔，以毫秒计

//创建定时器timer, 每隔1秒调用一次函数function

```
var timer = setInterval( function(){},1000);
```

### 关闭定时器

#### `clearInterval(timer);`





## 示例

### 1. 实现秒表功能：显示时,分,秒，点击按钮开始计时



➤ innerHTML: 修改标签内的文本



## 作业

- 1, 将今天的内容代码敲一遍
- 2, 获取当前时间并按括号内的格式打印 (\*年\*月\*日 星期\* \*: \*: \*)
- 3, 实现秒表功能, 包含时,分,秒,毫秒, 点击按钮1开始计时, 点击按钮2结束计时。

00时00分28秒8

开始计时

结束计时

- 4, 判断两个日期相差的天数
- 5, 使用定时器实现进度条.

PS: 获取标签对象, 改变style属性的width值

```
var proBar = document.getElementById("progressBar");  
proBar.style.width = barW + "px";
```

进度条



23%

