

JavaScript 对象编程

学习目标

- (1) 理解 JavaScript 中的对象。
- (2) 掌握 JavaScript 的 Date、Math、String 等对象。
- (3) 掌握 JavaScript 的数组声明以及使用方法。
- (4) 掌握正则表达式的属性和方法。
- (5) 掌握自定义对象的使用方法。

JavaScript 是一种基于对象的语言,它支持 3 种对象:内置对象、用户自定义对象以及浏览器对象。常用的内置对象包括 Date 对象、Math 对象、String 对象、Array 对象及 RegExp 对象等。下面将介绍常用内置对象的属性和方法及如何使用 JavaScript 的内置对象。

任务 3.1 实现京东秒杀效果



任务描述

在各大电商平台,常常能看到秒杀活动,下面以京东平台为例,每天固定时间开展不同商品的秒杀,6~8 点为 8 点场,后面每隔两小时做一场秒杀,每场时长两小时,直到晚上 0 点结束,如图 3-1 所示。



任务分析

科技创新给人们的生活带来了许多便利,通过电商平台足不出户就可以买到生活必需品,各电商平台也常常开展秒杀活动,不仅惠及普通百姓,也提高商家的商品销量,达到双赢效果。京东平台每天就有固定场的秒杀,这个特效的实现可以通过以下 4 个步骤完成。

- (1) 完成如图 3-1 所示京东秒杀页面设计,标识显示时间的位置。
- (2) 定义函数,使用日期和时间对象获取客户端时间,并设置场次结束时间。
- (3) 根据时间差换算成时、分、秒。
- (4) 使用定时函数,每隔 1 秒重新调用一次函数。

日期和时间对象(Date 对象)主要提供了获取和设置日期与时间的方法,JavaScript 中提供了两个定时器函数: `setTimeout()` 和 `setInterval()`。



图 3-1 京东秒杀效果

3.1.1 Date 对象的创建

要使用 Date 对象,必须先使用 new 运算符创建它,Date 对象的构造函数通过可选的参数,可生成过去、现在和将来的 Date 对象,创建 Date 常见方式有 3 种。

1. 不带参数

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

创建一个含有系统当前日期和时间的 Date 对象变量 myDate。

2. 创建一个指定日期的 Date 对象

```
var myDate = new Date("2022/10/01");
```

使用代表日期和时间的字符串创建一个特定日期的 Date 对象,上述语句创建了 2022 年 10 月 1 日的 Date 变量 myDate。

3. 创建一个指定时间的 Date 对象

```
var myDate = new Date(2022, 6, 1, 10, 30, 20);
```

语句创建了一个包含确切日期和时间的 Date 变量 myDate,即 2022 年 6 月 1 日 10 点 30 分 20 秒。

3.1.2 Date 对象的常用方法

Date 对象提供了很多操作日期和时间的方法,方便程序员在脚本开发过程中简单、快捷地操作日期和时间。表 3-1 列出了其常用的方法。

表 3-1 Date 对象常用方法

方法名	描述
getFullYear()	返回年份数
getMonth()	返回月份数(0~11)



续表

方 法 名	描 述
getDate()	返回日期数(1~31)
getDay()	返回星期数(0~6)
getHours()	返回时数(0~23)
getMinutes()	返回分数(0~59)
getSeconds()	返回秒数(0~59)
getMilliseconds()	返回毫秒数(0~999)
getTime()	返回对应日期基线的毫秒
toLocaleString()	返回日期的字符串表示,其格式根据系统当前的区域设置来确定
setDate()	设置 Date 对象中月的某一天(1~31)
setFullYear()	设置 Date 对象中的年份(四位数字)
setHours()	设置 Date 对象中的小时(0~23)
setMilliseconds()	设置 Date 对象中的毫秒(0~999)
setMinutes()	设置 Date 对象中的分钟(0~9)
setMonth()	设置 Date 对象中的月份(0~11)
setSeconds()	设置 Date 对象中的秒钟(0~59)
setTime()	以毫秒设置 Date 对象

【实例 3-1】 模拟日历,每天打开这个页面都能定时显示年、月、日和星期几。

下面使用 Date 对象显示当前时间,实现代码如下:

```
//1. 创建一个当前日期的日期对象
var date = new Date();
//2. 获取其中的年、月、日和星期几
var year = date.getFullYear();
var month = date.getMonth();
var hao = date.getDate();
var week = date.getDay();
//console.log(year + " " + month + " " + hao + " " + week);
//3. 赋值给 div
var arr = [
    "星期日",
    "星期一",
    "星期二",
    "星期三",
    "星期四",
    "星期五",
    "星期六",
];
var div = document.getElementById("time");
div.innerText = "今天是:" + year + "年" + (month + 1) + "月" + hao + "日" + arr[week];
```

运行代码,效果如图 3-2 所示。

今天是: 2022年1月15日 星期六

图 3-2 模拟日历



3.1.3 任务实现

实现京东秒杀效果的操作步骤如下。

(1) 完成页面设计,美化页面,标识显示时间的位置,关键 HTML 代码和 CSS 样式如下。

HTML 代码:

```
<div class = "jingdong">
  <h2>京东秒杀</h2>
  <p><strong id = "clock">21:00 </strong>点场 距结束</p>
  <div class = "time">
    <span id = "shi"></span>
    <span id = "fen"></span>
    <span id = "miao"></span>
  </div>
</div>
```

CSS 样式:

```
<style>
  .jingdong{
    width: 190px;
    height: 260px;
    padding-top: 10px;
    background-image: url(images/jingdong.png);
    margin: 50px auto;
  }
  h2{
    margin-top: 31px;
    font-size: 30px;
  }
  h2,p{
    color: #fff;
    font-weight: 700;
    text-align: center;
  }
  p{
    margin-top: 80px;
    font-size: 16px;
  }
  p strong{
    font-size: 18px;
  }
  .time{
    position: relative;
    width: 130px;
    height: 30px;
    margin: 0 auto;
  }
  span{
```



```
    position: relative;
    display: inline-block;
    width: 30px;
    height: 30px;
    line-height: 30px;
    text-align: center;
    background-color: #2f3430;
    color: #fff;
    margin: 0 5px;
    font-weight: bold;
    font-size: 18px;
}
span:nth-child(-n + 2)::after{
    content: ":";
    display: inline-block;
    position: absolute;
    bottom: 0;
    left: 27px;
    width: 20px;
    height: 30px;
    color: #fff;
    font-size: 18px;
}
</style>
```

(2) 添加脚本, 定义函数, 实现秒杀效果, 代码如下:

```
<script>
function countTime(){
    var now = new Date();
    var nowHour = now.getHours();
    var Session = "";
    if(nowHour >= 6){
        var end = new Date();
        end.setMinutes(0);
        end.setSeconds(0);
        //根据当前时间判断是哪场秒杀
        if (nowHour >= 6 && nowHour < 8){
            Session = "6:00";
            //确定结束时间
            end.setHours(8);
        }
        if (nowHour >= 8 && nowHour < 10){
            Session = "8:00";
            end.setHours(10);
        }
        if (nowHour >= 10 && nowHour < 12){
            Session = "10:00";
            end.setHours(12);
        }
        if (nowHour >= 12 && nowHour < 14){
```



```
        Session = "12:00";
        end.setHours(14);
    }
    if (nowHour >= 14 && nowHour < 16){
        Session = "14:00";
        end.setHours(16);
    }
    if (nowHour >= 16 && nowHour < 18){
        Session = "16:00";
        end.setHours(18);
    }
    if (nowHour >= 18 && nowHour < 20){
        Session = "18:00";
        end.setHours(20);
    }
    if (nowHour >= 20 && nowHour < 22){
        Session = "20:00";
        end.setHours(22);
    }
    if (nowHour >= 22){
        Session = "22:00";
        end.setHours(0);
    }
} else {
    Session = "暂时没有秒杀";
}
document.getElementById("clock").innerHTML = Session;

var lag = (end - now) / 1000;           //计算时间差
var hour = parseInt(lag / (60 * 60));  //将时间差转换成小时数
var minute = Math.floor((lag / 60) % 60); //将时间差转换成分钟数
var second = Math.floor(lag % 60);     //将时间差转换成秒数
//若时分秒小于10,则在其前面补0显示
hour = hour < 10 ? "0" + hour : hour;
minute = minute < 10 ? "0" + minute : minute;
second = second < 10 ? "0" + second : second;
document.getElementById("shi").innerHTML = hour;
document.getElementById("fen").innerHTML = minute;
document.getElementById("miao").innerHTML = second;
setTimeout("countTime()", 1000);
}
//页面加载时调用函数
window.onload = countTime;
</script>
```

运行代码,效果如图 3-3 所示。



图 3-3 京东秒杀效果



任务 3.1 实现京
东秒杀效果微课

任务 3.2 制作随机选号页面



任务描述

假设班上有 60 名同学,现制作一个提问选号器,如图 3-4 所示。单击“开始”按钮在页面随机显示 1~60 的学号,单击“停止”按钮在页面显示选中学号。

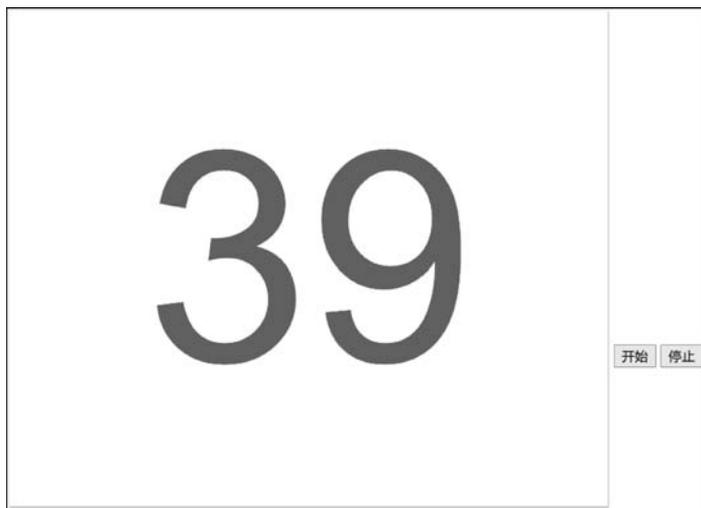


图 3-4 提问选号器页面



任务分析

可以通过以下步骤实现随机选号页面的制作。

- (1) 产生 1~60 的随机整数,并在页面显示。
- (2) 单击“开始”按钮时使用定时函数隔 60 毫秒产生一个随机整数。
- (3) 单击“停止”按钮时清除定时函数。

现在问题的关键是产生 1~60 的随机整数。产生随机整数需要用到 Math 对象。Math 对象包含用于各种数学运算的属性和方法。

3.2.1 Math 对象的常用属性

Math 对象的内置方法可以在不使用构造函数创建对象时直接调用。使用 Math 对象的属性时,常用格式如下:

Math.属性

Math 对象的常用属性见表 3-2。

表 3-2 Math 对象的常用属性

属 性 名	描 述
E	自然对数的底数
LN2	2 的自然对数
LN10	10 的自然对数
LOG2E	以 2 为底 e 的对数
LOG10E	以 10 为底 e 的对数
PI	圆周率
SQRT1_2	1/2 的平方根
SQRT2	2 的平方根

3.2.2 Math 对象的常用方法

Math 对象的方法是一些十分有用的数学函数,常用的方法见表 3-3。

表 3-3 Math 对象的常用方法

方 法	说 明
ceil(数值)	大于等于该数值的最小整数
floor(数值)	小于等于该数值的最大整数
min(数值 1,数值 2)	最小值
max(数值 1,数值 2)	最大值
pow(数值 1,数值 2)	数值 1 的数值 2 次方
random()	0~1 的随机数
round(数值)	最接近该数值的整数
sqrt(数值)	开平方根

还有 abs、sin(弧度)、cos、tan、asin、acos、atan、exp、log 等



例如,计算 $\cos(\text{PI}/6)$,可以写成: `Math.cos(Math.PI/6)`。

3.2.3 任务实现

完成如图 3-4 所示随机选号器制作的代码如下(页面样式省略):

```
<!DOCTYPE html >
<html >
<head >
  <meta charset = "utf - 8" />
  <title>选号器</title >
  <script language = "javascript" type = "text/javascript">
    var timer;
    function startScroll(){
      var num = Math.floor(Math.random() * 60 + 1);
      document.getElementById("mytext").value = num;
      // 开始定时器前,将前一个定时器清除
      clearTimeout(timer);
      timer = setTimeout("startScroll()", 60);
    }
    function stopScroll(){
      clearTimeout(timer);
    }
  </script >
</head >
<body >
  <center >
    <form name = "myForm">
      <input id = "mytext" type = "text" value = "0" class = "inputTxt" />
      <input name = "start" type = "button" value = "开始" onclick = "startScroll()" />
      <input name = "stop" type = "button" value = "停止" onclick = "stopScroll()" />
    </form >
  </center >
</body >
</html >
```



任务 3.2 制作随机选号页面微课

任务 3.3 制作简单的焦点图效果



任务描述

焦点图效果是各大网站常用的效果,下面利用数组实现简单的焦点图效果,如图 3-5 所示,页面上 5 张图片 2 秒轮换显示,单击向右图片实现播放下一张图片,图片向后继续 2 秒轮换显示,单击向左图片实现播放上一张图片,图片向前继续 2 秒轮换显示。



图 3-5 简单焦点图效果



任务分析

实现简单焦点图效果可以采用以下步骤。

- (1) 设计 HTML 页面,应用 CSS 美化页面。
- (2) 定义数组,将轮换显示的图片地址保存到数组中。
- (3) 定义两个全局变量,一个变量用于控制定时器,另一个变量用于控制数组下标。
- (4) 定义函数实现图片的轮换显示。在函数中改变图片的地址,使用定时器函数,2 秒更换图片地址,实现图片的轮流显示。
- (5) 单击上一张或下一张按钮时将定时器清除,再重新调用图片轮换显示函数。

数组是包含基本数据类型和组合数据类型的有序序列,JavaScript 中数组也是最常用的对象之一,下面介绍数组的创建及其常用属性和方法。

3.3.1 数组的创建

数组是值的有序集合,由于弱类型的原因,JavaScript 中数组十分灵活、强大,不像 Java 等强类型高级语言,数组只能存放同一类型或其子类型元素,JavaScript 在同一个数组中可以存放多种类型的元素,而且长度是可以动态调整的,可以随着数据增加或减少自动对数组长度进行更改。

1. 创建数组

在 JavaScript 中,可以使用构造函数创建数组,也可以直接使用方括号创建数组,具体有以下几种创建方法。

- (1) 使用无参构造函数,创建一空数组。

```
var arr = new Array();
```

- (2) 一个数字参数构造函数,指定数组长度。

```
var arr = new Array(5);
```

表示创建一个长度为 5 的数组,由于数组长度可以动态调整,作用并不大。

- (3) 带有初始化数据的构造函数,创建数组并初始化参数数据。

```
var arr = new Array("HTML", "JavaScript", "DOM")
```

- (4) 使用方括号,创建空数组,等同于调用无参构造函数。

```
var arr = [];
```