



## 第二十五天到第二十七天 倒数开始 滴答滴 滴答滴

普通

👤 Varsha (/mentor/detail/id/2) | 🏠 零基础学院 (/college/detail/id/5)

开始学习

人数 有 300 人在学习该课程，有 254 人已经完成该课程

时间 平均用时 2.0 天

关键词 JavaScript

### 课程概述

作业提交截止时间：09-01

## 第二十五天到第二十七天 倒数开始 滴答滴 滴答滴

### 课程目标

学习 JavaScript 的函数，学习如何使用和封装函数，学习Date对象

### 课程描述

### 阅读

可能大部分同学已经理解了函数是什么概念，不过对于一些新入门的同学，我们不妨再来仔细阅读一下什么是函数。

- W3School 函数 ([http://www.w3school.com.cn/js/js\\_functions.asp](http://www.w3school.com.cn/js/js_functions.asp))
- MDN ([https://developer.mozilla.org/zh-CN/docs/Learn/JavaScript/Building\\_blocks/Functions](https://developer.mozilla.org/zh-CN/docs/Learn/JavaScript/Building_blocks/Functions))

## 编码

我们假设要在某种特殊的机器人上写代码，但是这个机器人本身只实现了以下两个方法（函数）：

- Go，表示向当前方向前进一步
- TurnLeft，表示向左转

那我现在需要你做的任务是向前走两步，然后向左转，再向前走一步，所以代码是：

```
Go();
Go();
TurnLeft();
Go();
```

好，看上去没什么问题，我们加大难度，我们希望现在不向左转了，而是向右转，那么如何实现向右转了，我们可以通过三次向左转来实现

```
Go();
Go();
TurnLeft();
TurnLeft();
TurnLeft();
Go();
```

接下来，我重新定需求，我需要这个机器人，向前一步，然后向右转，向前两步，再向右转，向前一步，再向右转，向前三步，再向右转，那代码变成：

```
Go();
TurnLeft();
TurnLeft();
TurnLeft();
Go();
Go();
TurnLeft();
TurnLeft();
TurnLeft();
Go();
TurnLeft();
TurnLeft();
TurnLeft();
Go();
Go();
Go();
TurnLeft();
TurnLeft();
TurnLeft();
```

这样看就比较长了，怎么解决呢？我们可以把一些重复的，可复用可抽象的代码封装成一个函数，我们现在新增一个函数叫做 TurnRight()，虽然机器人不直接支持向右转，但是我们通过一个封装了3次向左转的函数，实现了这个功能：



```
function TurnRight() {
    TurnLeft();
    TurnLeft();
    TurnLeft();
}
Go();
TurnRight();
Go();
Go();
TurnRight();
Go();
TurnRight();
Go();
Go();
Go();
TurnRight();
```

这样我们的代码就精简了许多，接下来需求变成了前进4步，向右转，前进8步向右转，前进20步向右转.....不想写了，大家应该能想到我们需要写很多的Go()，那怎么封装呢？难道我们封装很多 Go4Steps(), Go8Steps()吗？显然这样做只是用一种不优雅的方式代替了另外一种不优雅的方式。我们会发现Go多少步，行为是一致的，只是执行的次数不同，所以我们可以新增一个函数，把要执行的步数作为参数传给这个函数，如：

```
function GoSteps(n) {
    while (n-->0) {
        Go();
    }
}
```

这样我们的代码就会简化很多，当然上面的GoSteps函数目前是不严谨的，我们必须考虑更多的问题，比如传入参数的正确性和合法性，比如传入的参数是不是数字，是不是正整数，现在请你自己写一个GoSteps函数，然后跑通下面的测试用例。

```
function Go() {
    console.log("Go");
}

function GoSteps(n) {
    .....
}

GoSteps(10); // Go 10次
GoSteps(1); // Go 1次
GoSteps(); // Go 1次，认为缺少参数时，默认参数为1
GoSteps(0); // 0次
GoSteps(-1); // 0次
GoSteps(1.4); // Go 1次
GoSteps(1.6); // Go 1次
GoSteps(-1); // 0次
GoSteps(true); // Go 1次
GoSteps(false); // 0次
GoSteps("Test"); // 0次
GoSteps(NaN); // 0次
GoSteps(null); // 0次
```



## 阅读

接下来我们看一个复杂一些的问题，不过在这之前，我们先学习一下 Date 对象，来帮助我们接下来的练习。

- W3School Date ([http://www.w3school.com.cn/js/js\\_obj\\_date.asp](http://www.w3school.com.cn/js/js_obj_date.asp))

然后再复习一下 `SetTimeout` 和 `setInterval`

- W3School 定时 ([http://www.w3school.com.cn/js/js\\_timing.asp](http://www.w3school.com.cn/js/js_timing.asp))

## 编码

我们现在来做一个最简单的时钟，通过小练习来学习 `Date`，复习定时，然后再练习一下函数的封装  
具体需求如下：

- 在页面中显示当前日期及时间，按秒更新
- 格式为 YYYY 年 MM 月 DD 日 星期 D HH:mm:ss
- 注意位数的补齐，比如：
  - 假设时间为2008年10月10日星期一的12点12分12秒，显示2008年10月10日星期一 12:12:12
  - 假设时间为2008年1月1日星期一的3点2分2秒，显示2008年01月01日星期一 03:02:02

编码过程中，我们希望你注意对函数的封装，尽量让一个函数就做一个事情，而不要把所有的功能、代码揉在一起：

- 封装一个函数，来根据某个日期返回这一天是星期几
- 封装一个函数，把月、日、小时等出现个位数的情况前面补充0，补充为两位，比如1变为01
- 封装一个函数，把最后的日期时间，按照要求的格式进行包装
- 可能不止上面这些，尽可能地进行功能的解耦和拆解

## 编码

完成上面需求后，现在需求做一些小的变更

- 输出格式变为：2008-10-10 Monday 07:10:30 PM

## 编码

基于上面的需求，要求，同时在页面上，输出两种格式的日期时间  
希望通过上面的练习，让你体会函数的概念、作用和价值

## 编码

现在我们要做一个稍微复杂的东西，如下HTML，有一堆Select用于选择日期和时间，在选择后，实时在 id 为 `result-wrapper` 的 `p` 标签中显示所选时间和当前时间的差值。



```

<select id="year-select">
  <option value="2000">2000</option>
  <option value="2001">2001</option>
  <option value="2002">2002</option>
  .....
  <option value="2032">2002</option>
</select>

<select id="month-select">
  <option value="1">1</option>
  <option value="2">2</option>
  .....
  <option value="12">12</option>
</select>

<select id="day-select">
  <option value="1">1</option>
  <option value="2">2</option>
  .....
  <option value="31">31</option>
</select>

<select id="hour-select">
  <option value="0">00</option>
  <option value="1">01</option>
  .....
  <option value="23">23</option>
</select>

<select id="minite-select">
  <option value="0">0</option>
  <option value="1">1</option>
  .....
  <option>59</option>
</select>

<select id="second-select">
  <option value="0">0</option>
  <option value="1">1</option>
  .....
  <option>59</option>
</select>

<p id="result-wrapper">现在距离 2001年1月1日星期X HH:MM:SS 还有 X 天 X 小时 X 分 X 秒
</p>

```

- 使用上方的HTML结构（可以根据需要自行微调）为基础编写JavaScript代码
- 当变更任何一个select选择时，更新 result-wrapper 的内容
- 当所选时间早于现在时间时，文案为 现在距离 "所选时间" 已经过去 xxxx
- 当所选时间晚于现在时间时，文案为 现在距离 "所选时间" 还有 xxxx
- 注意当前时间经过所选时间时候的文案变化
- 注意选择不同月份的时候，日期的可选范围不一样，比如1月可以选31天，11月只有30天，注意闰年



- 同样，需要注意，通过优雅的函数封装，使得代码更加可读且可维护

## 进阶任务

如果你很快就完成上面的任务，可以去LeetCode上去多进行一些练习。

## 提交

把你今天觉得做得最好的代码放在Github后进行提交

## 总结

依然把今天的学习用时，收获，问题进行记录

## 下一个任务预告

接下来我们会做很多综合性的小练习，来学习如何用HTML，CSS和JavaScript解决一些小问题

## 课程优秀学习笔记

暂无优秀学习笔记~

IFE微信公众号: baidu\_ife    IFE微博: Baidu前端技术学院 (<http://weibo.com/u/5568860641?topnav=1&wvr=6&topsug=1>)

友情链接:    百度EFE (<http://efe.baidu.com>)    百度校园 (<http://campus.baidu.com/>)    百度校园招聘  
(<http://talent.baidu.com/external/baidu/campus.html>)    百度技术学院 (<http://bit.baidu.com/?fr=ife/>)

资源:    IFE 2015 (<https://github.com/baidu-ife>)    IFE 2016 (<http://ife.baidu.com/2016/static/index.html>)    IFE 2017  
(<http://ife.baidu.com/2017>)

©2017 Baidu 使用百度前必读 (<http://www.baidu.com/duty/>) 意见反馈 (<http://jianyi.baidu.com>) 京ICP证030173号

