## if 语句

我们已经在第三章中了解了 JavaScript 中数据的使用，掌握到的预备知识是我们具备了随心所欲操控数据的能力，在这一章中我们将学习 JavaScript 中的*语句*，以编写出“真正的”程序，并逐步将我们头脑中的逻辑，转化为程序真切的运行过程。

在前面的运行示例中，我们接触到的都是顺序结构，即程序从第一条语句，逐行执行到结尾，而在这一章中我们将了解到*分支结构*和*循环结构*。

我们首先要学习的是 *if 语句*。它用于实现分支结构。

想象一下，当我们要去买芹菜的时候，我们会执行这样的过程：

1. 出门，找到菜市场
2. 挑选芹菜
3. 付钱，将菜带回家。

这就是我们所说的顺序结构。但实际上，我们买菜可能会经历这样的过程：

1. 出门，找到菜市场
2. 寻找卖芹菜的摊位

* 如果有卖芹菜的
  1. 买一斤芹菜
  2. 称重，付款，回家

如果没有卖芹菜的

1. 看看有没有其他适合的蔬菜
2. 称重，付款，回家

以上过程的结构称为*分支结构*，我们可以用 if 语句来在程序中实现分支结构，它的*语法*是这样的：

if (条件) {  
 执行语句  
}

一条 if 语句以关键字 if 开头，其后跟随一对括号，括号中是一个逻辑表达式，用于表示执行语句的条件。

例如：

let a = 4;  
if (a > 3) {  
 alert("a 大于 3");  
}

这段代码中的 if 语句的条件是 a > 3，所执行的语句是 alert("a 大于 3"); 。显然，a 大于 3，因此条件成立（逻辑表达式得到 true），那么执行大括号中的语句——显示出 "a 大于 3" 这行内容。

if 语句只管理它大括号中的内容，其他部分不受条件影响。

倘若我们要考虑两种情况（如示例中的“有卖芹菜的摊位”与“没有卖芹菜的摊位”），那么我们可以使用 if 语句的另一种形式。

if (条件) {  
 语句1  
} else {  
 语句2   
}

这种形式的作用是：如果条件成立，那么执行代码1中的内容，否则，执行代码2。

else 关键字所*引导*的内容称为 *else 子句*，它用于说明“条件不成立”这一情况。

示例：

let a = 4;  
if (a > 5) {  
 alert("a 大于 5");  
} else {  
 alert("a 小于等于 5");  
}

在示例中，由于 a 的值为 4，if 语句的判断条件不成立（得到 false），所以不会显示 "a 大于 5"，而是执行 else 子句中的语句，显示 "a 小于等于 5"。如果 a 的值大于 5，符合判断条件的话，那么就会显示 "a 大于 5"。

我们似乎已经能够处理条件成立和不成立两种情况了。但现实生活比这复杂的多，我们可能会遇到许多其他情况，需要对每种情况一一加以判断并分别作出相应决定。在 JavaScript 中，我们可以使用 if 语句的高级形式，像这样：

let score = 85;  
if (score >= 90) {  
 alert("优秀");  
} else if (score >= 80) {  
 alert("良好");  
} else if (score >= 60) {  
 alert("及格");  
} else {  
 alert("不及格");  
}

上述代码模拟了一种常见情景：将成绩分为不同层次，并作出对应通知。

它的执行过程会先从第一个条件开始，如果条件成立，那么执行其后的语句；如果条件不成立，那么尝试判断第二个条件，若成立则执行相应语句，否则判断第三个条件……直到最后一条 else 子句，即当所有可能条件都不成立时，执行其中的语句。在上述代码中，分数大于等于 80，我们便得到了“良好”。

我们在第三章已经阐述过，JavaScript 会将一些值视为“真”，另一些视为“假”，以此进行逻辑运算，而不仅仅是局限于布尔值。同样，在 if 语句的条件并不要求得到一个布尔值，只要得到的值被视作“真”，就会执行相应语句。

因此，假如我们有如下代码：

// 判断一个数字是奇数还是偶数。  
// 如果它除以 2 的余数为 0，即能被 2 整除，那么是偶数。  
// 否则为奇数。  
let number = 100;  
if (number % 2 === 0) {  
 alert("偶数");  
} else {  
 alert("奇数")  
}

当条件的值为 0 时，它会被当做假，那么我们可以用更方便的形式书写条件：

let number = 100;  
if (number % 2) { // 余数不为 0，会当做“真”  
 alert("奇数");  
} else { // 余数为 0，会当做“假”  
 alert("偶数")  
}

if 语句是我们接触到的第一个 JavaScript 控制语句。我们可以用它来描述真实世界中的各类选择。

一条简洁明快的 if 语句，将一切分成了真假两个世界，中间隔着逻辑这条天河，它向何处流动，取决于你的思考。逻辑的伟力在于泾渭分明，逻辑的生命在于它所带来的不容逾越的秩序。

练习 4.1.1

1. 写一个幸运转盘的程序，每次根据一个随机数字的范围来决定颁发什么奖品。
2. 写一个程序，让用户输入出生年份，判断用户的生肖属相。如果不是一个合理的年份，就显示一个错误。