

## 第 3 章 程序基础知识

App Inventor 2 作为一种编程软件,它与其他程序设计软件一样,可以设置变量并进行数字、逻辑等运算,同时它提供循环、分支等结构让程序可以更加多样化,适合不同的设计要求,让 App 的程序设计有更加广泛的应用。

### 3.1 程序设计的几个概念

#### 1. 变量

(1) 变量概念源于数学,在计算机语言中能储存计算结果或能表示值的抽象概念。变量首先需要进行声明,其后可以通过变量名来进行访问。

(2) 声明原则和类型。变量名必须以字母或下划线打头,名字中间只能由字母、数字和下划线组成;不能以数字开头,不能使用中文。App Inventor 中变量类型有数字型(number)、文字型(text)、逻辑型(logic)和列表型(list)。声明变量时要初始化赋值。

(3) 变量包括局部(local)变量(只能在事件模块中调用)和全局(global)变量(在整个 App 中均可调用)。

定义变量的步骤如下。

(1) 定义一个全局变量,如图 3-1 所示。

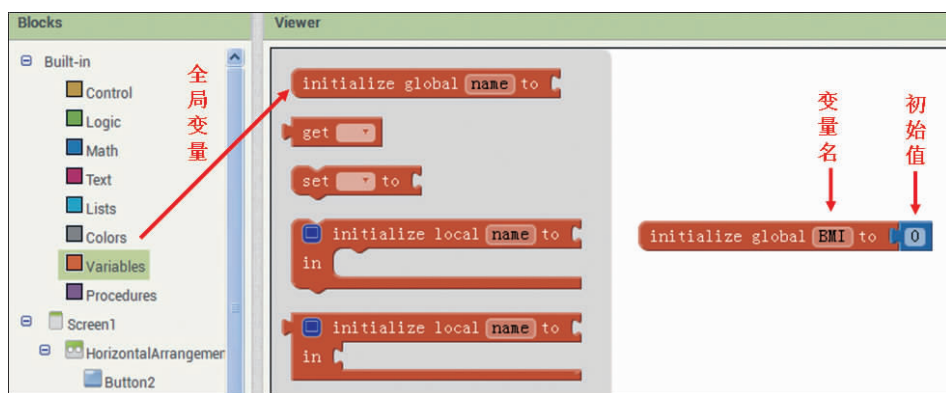


图 3-1 定义一个全局变量

(2) 定义一个局部变量,如图 3-2 所示。

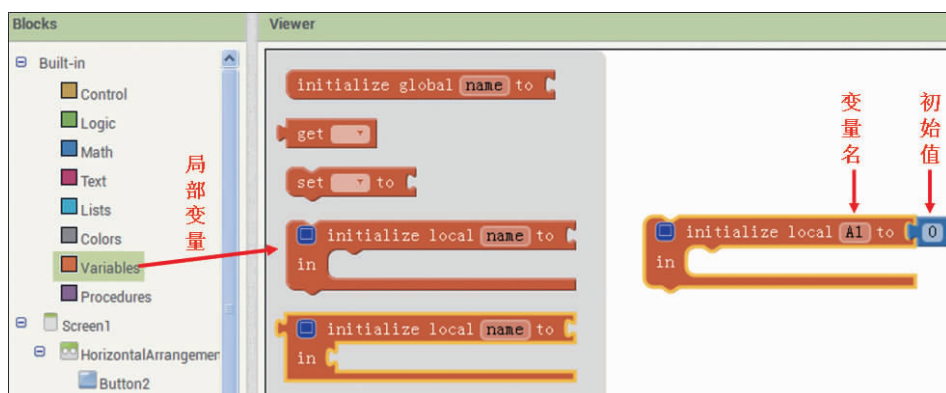


图 3-2 定义一个局部变量

## 2. 属性

组件的属性是指组件的大小、颜色、位置、形状等特性,属性可以在前端 UI 设计界面中进行设置,也可以在 Blocks 模块中设置。Blocks 模块中属性模块用绿色表示,如图 3-3 所示。

## 3. 事件

事件是可以被控件识别的操作,如单击“确定”按钮、选择某个单选按钮或者复选框。每一种控件有自己可以识别的事件,如窗体的加载、单击、双击等事件,编辑框(文本框)的文本改变事件等。

Blocks 模块中事件为暗黄色马蹄形,如图 3-4 所示。

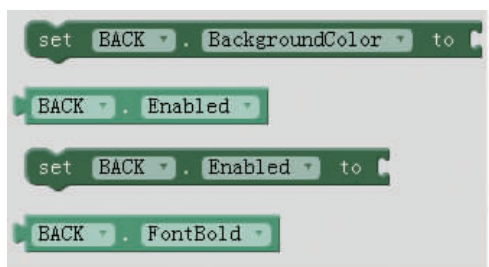


图 3-3 设置属性



图 3-4 事件设置

## 4. 方法

方法是指触发内部程序,如弹出对话框、隐藏键盘、保存数据等。

方法模块为紫色,如图 3-5 所示。

## 5. 过程

过程是存放在某个名称下的一系列块的组合,或者代码。在计算机科学中,过程也称为函数。在开发中如果需要反复使用同一个块集合,此时通过定义过程,可减少代码冗余。过程可以有返回值,也可以没有。一个过程可以没有或者有多个参数。一般来说,一





个过程完成一项功能,如图 3-6 所示。

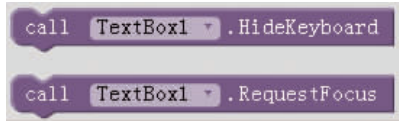


图 3-5 方法设置

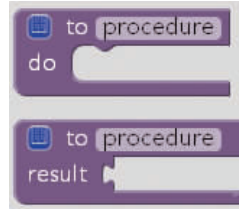


图 3-6 建立过程

### 6. 列表

列表(List)是 App Inventor 中 3 种类型变量之一。在 App Inventor 中列表是一个可以存放多个相同类型的元素的集合。

(1) 新建列表并增加数据,如图 3-7 所示。

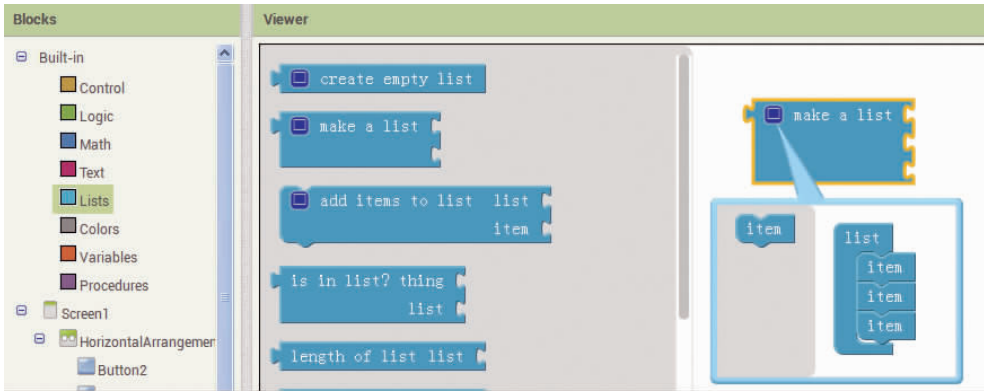


图 3-7 新建列表

(2) 创建一个全局列表,如图 3-8 所示。



图 3-8 全局变量为一列表

(3) 如果调用其中某一数据就要用到索引值,索引从 1 开始,索引编号不能超过列表数据项的总数,如图 3-9 所示。

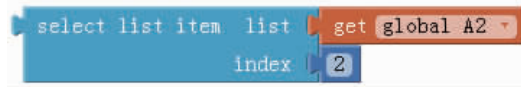


图 3-9 指定索引值

列表可以是一维的或二维的,它相当于其他编程语言的数组。在内存中,列表中的元



素是按先后顺序连续存放。列表的值通过列表名称及其索引值引用。列表索引从 1 开始,如果索引的值大于列表的长度系统就会报错。

一维列表如表 3-1 所示。

表 3-1 一维列表

索引	1	2	3
列表	Daive	Lisa	Tony

将一维列表存为变量,模块如图 3-10 所示。



图 3-10 变量为一维列表

二维列表如表 3-2 所示。

表 3-2 二维列表

索引	1	2
1	20151001	Daive
2	20151002	Lisa
3	20151003	Tony

将二维列表存为变量,模块如图 3-11 所示。

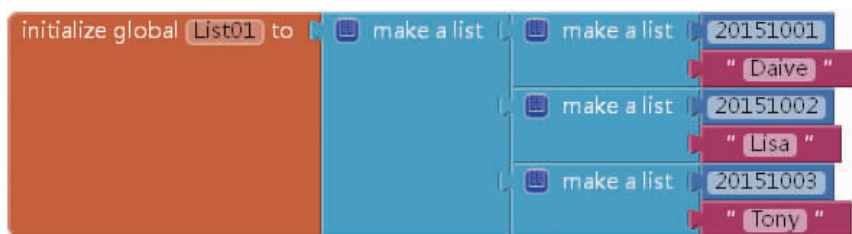


图 3-11 变量为二维列表

## 7. 循环结构

(1) for each from 模块,如图 3-12 所示。

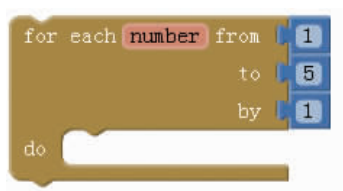


图 3-12 for each from 模块





这一模块中,from、to、by 分别连接变量 number 的初值、变量 number 的末值和每次循环的递增量,如图 3-13 和图 3-14 所示,连接到指定值并显示结果。

```

when Button_1 .Click
do
  initialize local A1 to 0
  in for each number from 1
    to TextBox_1 . Text
    by 1
  do
    set A1 to get number + get A1
  set Label_1 . Text to get A1

```

图 3-13 变量为局域变量一

(2) while 模块,如图 3-15 所示。

```

initialize global A2 to 0
when Button_2 .Click
do
  set global A2 to 0
  for each N2 from 1
    to TextBox_2 . Text
    by 1
  do
    set global A2 to get N2 + get global A2
  set Label_2 . Text to get global A2

```

图 3-14 变量为局域变量二

```

while test
do

```

图 3-15 while 模块

这一模块表示当条件满足时执行循环,test 连接循环的条件,如图 3-16 所示,指定数字进行累加并显示最后结果。

```

initialize global A3 to 0
initialize global N3 to 1
when Button_3 .Click
do
  while test
    get global N3 <= TextBox_3 . Text
  do
    set global A3 to get global N3 + get global A3
    set global N3 to get global N3 + 1
  set Label_3 . Text to get global A3

```

图 3-16 数字累加