



移动开发人才培养系列丛书

App Inventor

移动应用开发 | 标准教程

Building
Android App Using
App Inventor

瞿绍军 编著

- 获谷歌支持教育部产学合作专业综合改革项目资助
- 近 90 个教学案例，6 个综合项目
- 提供多媒体课件、源代码、教学视频等丰富配套资源



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



移动开发人才培养系列丛书

App Inventor

移动应用开发 | 标准教程

Building
Android App Using
App Inventor

瞿绍军 编著

人民邮电出版社

北京

图书在版编目 (C I P) 数据

App Inventor移动应用开发标准教程 / 瞿绍军编著

— 北京 : 人民邮电出版社, 2016.8

(移动开发人才培养系列丛书)

ISBN 978-7-115-42681-9

I. ①A… II. ①瞿… III. ①移动终端—应用程序—
程序设计—教材 IV. ①TN929.53

中国版本图书馆CIP数据核字 (2016) 第120919号

内 容 提 要

本书主要以 App Inventor 作为平台, 把抽象的计算思维具体化, 把复杂的编程思想形象化, 通过积木块的编程, 将计算思维无缝地融入到有趣的 App Inventor 小项目中, 极大地提升学生的学习兴趣。通过该课程的学习, 学生不仅会大大提高动手能力, 而且可以从中获得巨大的成就感, 最终爱上计算机科学。

全书共分 6 章, 各章节内容由浅入深、相互衔接。前 4 章为基础篇, 主要介绍 App Inventor 的开发环境搭建、界面和代码块的操作方法、两个简单而有趣的入门项目、App Inventor 编程基础、组件; 第 5 章介绍应用调试的方法; 第 6 章是进阶篇, 介绍了 6 个综合项目。全书共提供教学案例近 90 个。

本书配备了丰富的教学和学习辅助资料, 包括课件、教材中项目的源代码、高清授课视频、综合项目源文件和作业参考源码文件等。从 2015 年开始, 这些资料已经通过 Google 面向全国高校和中学开放, 并且每年都在不断更新完善 (分享网址: <http://pan.baidu.com/s/1sk30sxB>)。

本书依据 App Inventor 的最新版本编写。全书在全面介绍知识点的同时, 也提供多个案例供读者练习, 并将计算思维贯穿其中, 内容由易到难, 适合不同层次的读者。本书既可以作为移动开发课程的教学用书, 也可以作为手机应用开发者的参考和工具用书。

-
- ◆ 编 著 瞿绍军
责任编辑 刘 博
责任印制 沈 蓉 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 17.5
字数: 460 千字
-
- 2016 年 8 月第 1 版
2016 年 8 月河北第 1 次印刷

定价: 49.80 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广字第 8052 号

2006 年 3 月，美国卡内基·梅隆大学计算机系主任周以真（Jeannette M. Wing）教授在美国计算机权威期刊《Communications of the ACM》上给出了计算思维（Computational Thinking, CT）的定义。

计算思维是指运用计算机科学的基础概念去求解问题、设计系统和理解人类行为。计算思维涵盖了反映计算机科学广泛性的一系列思维活动。计算思维的本质是抽象(Abstraction)和自动化(Automation)。它是如同所有人都具备的“读、写、算”（简称 3R）能力一样，成为适合于每个人的一种普遍的认识和一类普适的技能。计算思维体现了面向问题解决的系列观点和方法，有助于更加深刻地理解计算的本质和计算机求解问题的核心思想。

中国工程院院士李国杰任组长的中国科学院信息领域战略研究组在其撰写的《中国至 2050 年信息科技发展路线图》中对“计算思维”给予了足够的重视，认为计算思维的培育是克服“狭义工具论”的有效途径，是解决其他信息科技难题的基础。

近年来，全球很多组织和机构在通过各种途径在大到大学生的课程，小到中小学生的课程推广和培养学生们的计算思维能力。一批应用随之诞生，比如 Scratch、机器人编程、VB（Visual Basic）、App Inventor 等。其中，App Inventor 是近两年发展最为迅速的应用。

App Inventor 是一个基于网页开发 Android 移动应用程序的平台。借助 App Inventor 可以把抽象的计算思维具体化，把复杂的编程思想形象化，通过积木块的编程和将计算思维无缝地融入到一个个有趣的 App Inventor 小项目中，可极大地提升学生的学习兴趣。

虽然通过 App Inventor 编写的应用程序或许不是很完美，但它们却是普通人都能编写的，而且通常是在几分钟内就可完成。学生通过对 App Inventor 的学习，可快速了解到软件设计与开发的基本知识，掌握解决问题的方法，训练计算思维能力，并可在短时间内将自己的点子变成作品，不仅大大提高了动手能力，而且从中获得了巨大的成就感，最终促使学生爱上编程和计算机科学。

App Inventor 在 2010 年 12 月被正式对外发布。2013 年 12 月，麻省理工学院（MIT）发布了 App Inventor 2。截至 2016 年 3 月底，这个软件的全球用户达到 460 万人，注册的国家数达到 195 个，作品数量达到 1400 多万个。在 2012 年 11 月 15 日-16 日举行的“2012 年 Google 中国教育高峰会”上，哈尔艾贝尔森（Hal Abelson）教授针对 App Inventor 进行专题演讲及介绍，与会的部分教师表现出极大的兴趣。最终在 Google 的支持下 App Inventor 被迅速引入国内。2013 年上半年，国内举行了针对大学教师的师资培训班。随后，MIT、广州市教育信息中心和华南理工大学通过三方合作，在广州搭建了国内的服务器。在 Google 的 CS4HS（Computer Science for High School）项目支持下，2014 年，全国开展了 13 期面向中小学信息技术教师的 App Inventor 师资培训班；2015 年，全国开展了 20 期面向中小学信息技术教师的 App Inventor 师资培训班和 1 期面向种子培训学校的师资培训班；2016 年，全国将开展 21 期面向中小学信息技术教师的 App Inventor 师资培训班和 1 期面向种子培训学校的师资培训班。2014 年和 2015 年举行的 App Inventor 应用开发全国中学生挑战赛，2016 年将继续举办。2015 年 6 月，广州举行了以 App Inventor 为主题的“教育与计算思维教育国际研讨会”。目前，App Inventor 在国内已经走进众多中学、中职、高职和大学课堂。

本书是一本通过讲解知识点并将其与案例相结合的方式引导学生掌握移动应用程序开发方

法的教材。在编写过程中，力求用简洁、通俗的语言来描述复杂的概念，做到深入浅出、循序渐进，从而使学生能理解计算思维的真正内涵和体会到学习编程的乐趣。

本书的出版得到了“谷歌信息技术有限公司 2014 校企合作专业综合改革项目移动应用课程建设项目‘App Inventor 移动应用开发’”和“湖南师范大学教学改革研究项目‘程序设计类课程实践教学体系、内容、方法和手段改革的研究与实践’”的资助。

为方便教师的教学，本书配有丰富的电子资源，包括课件、教材中项目的源代码、高清授课视频、综合项目源文件和作业参考源代码文件等（分享网址：<http://pan.baidu.com/s/1sk30sxB>）。

您在使用过程中有任何疑问，可发邮件与作者联系（E-mail: powerhope@163.com）。

编 者

2016 年 4 月于长沙岳麓山

目 录

第 1 章 App Inventor 入门.....1

- 1.1 App Inventor 简介.....1
- 1.2 App Inventor 能做什么.....1
 - 1.2.1 玩.....2
 - 1.2.2 建立原型.....2
 - 1.2.3 构建个性化应用.....2
 - 1.2.4 开发完整的应用.....2
 - 1.2.5 教与学.....2
- 1.3 App Inventor 开发环境搭建.....2
 - 1.3.1 系统要求.....2
 - 1.3.2 服务器地址.....3
 - 1.3.3 三种测试应用的方法.....3
- 1.4 App Inventor 2 界面.....11
 - 1.4.1 项目管理界面.....11
 - 1.4.2 组件设计界面.....12
 - 1.4.3 逻辑设计界面.....13
 - 1.4.4 代码块的操作方法.....15
 - 1.4.5 Gallery.....20
- 1.5 实验.....22

第 2 章 创建几个简单的项目.....23

- 2.1 案例 1: HelloPurr, 你好猫猫.....23
 - 2.1.1 新建项目.....23
 - 2.1.2 界面设计.....24
 - 2.1.3 添加组件行为.....27
 - 2.1.4 测试应用.....28
 - 2.1.5 添加震动效果.....29
 - 2.1.6 添加摇晃手机发出猫叫功能.....30
 - 2.1.7 打包和下载应用.....31
- 2.2 案例 2: 8-Ball, 魔术 8 球.....32
 - 2.2.1 新建项目.....33
 - 2.2.2 界面设计.....33
 - 2.2.3 添加组件行为.....34
 - 2.2.4 实现“单击按钮输出一个预言”
功能.....36

- 2.2.5 实现“晃动手机输出一个预言”的
功能.....39
- 2.2.6 打包和下载应用.....39
- 2.3 实验.....39

第 3 章 App Inventor 编程基础.....40

- 3.1 变量块 (Variables blocks).....40
 - 3.1.1 全局变量的定义和使用.....40
 - 3.1.2 局部变量的定义和使用.....42
 - 3.1.3 带有返回值的局部变量块的使用.....43
- 3.2 控制块 (Control blocks).....43
 - 3.2.1 选择.....44
 - 3.2.2 循环.....47
 - 3.2.3 条件返回.....50
 - 3.2.4 带有返回值的执行模块.....51
 - 3.2.5 求值但忽视结果.....51
 - 3.2.6 打开屏幕.....52
 - 3.2.7 打开屏幕并传值.....53
 - 3.2.8 其他控制块.....53
- 3.3 逻辑块 (Logic blocks).....53
- 3.4 数学块 (Math blocks).....54
- 3.5 文本块 (Text blocks).....58
- 3.6 列表块 (Lists blocks).....63
- 3.7 颜色块 (Colors blocks).....71
- 3.8 过程块 (Procedures blocks).....73
- 3.9 递归.....74
- 3.10 组件块.....75
- 3.11 任意组件块.....75
- 3.12 项目: 一元二次方程求根.....75
- 3.13 实验.....78

第 4 章 组件.....79

- 4.1 Screen.....80
- 4.2 用户界面组件 (User Interface
components).....81
 - 4.2.1 按钮 (Button).....82

4.2.2 文本输入框 (TextBox)	83	4.6.4 位置传感器 (LocationSensor)	155
4.2.3 列表显示框 (ListView)	85	4.6.5 条码扫描器 (BarcodeScanner)	157
4.2.4 日期选择框 (DatePicker)	86	4.6.6 近场通信 (NFC)	158
4.2.5 时间选择框 (TimePicker)	88	4.6.7 近距离传感器 (ProximitySensor)	158
4.2.6 复选框 (CheckBox)	88	4.6.8 陀螺仪传感器 (GyroscopeSensor)	159
4.2.7 标签 (Label)	90	4.7 社交应用组件 (Social Components)	160
4.2.8 列表选择框 (ListPicker)	90	4.7.1 联系人选择框 (ContactPicker)	161
4.2.9 滑动条 (Slider)	92	4.7.2 电话号选择框 (PhoneNumberPicker)	162
4.2.10 密码输入框 (PasswordTextBox)	94	4.7.3 电话拨号器 (PhoneCall)	163
4.2.11 对话框 (Notifier)	94	4.7.4 短信收发器 (Texting)	164
4.2.12 图像 (Image)	97	4.7.5 邮箱地址选择框 (EmailPicker)	166
4.2.13 Web 浏览框 (WebViewer)	97	4.7.6 信息分享器 (Sharing)	166
4.2.14 下拉框 (Spinner)	100	4.7.7 推特客户端 (Twitter)	168
4.3 界面布局组件 (Layout components)	103	4.8 数据存储组件 (Storage)	172
4.3.1 水平布局 (HorizontalArrangement)	103	4.8.1 文件管理器 (File)	172
4.3.2 垂直布局 (VerticalArrangement)	104	4.8.2 微数据库 (TinyDB)	176
4.3.3 表格布局 (TableArrangement)	105	4.8.3 网络微数据库 (TinyWebDB)	179
4.4 多媒体组件 (Media components)	105	4.8.4 融合图表控制组件 (FusiontableControl)	182
4.4.1 音效 (Sound)	106	4.9 通信连接组件 (Connectivity)	182
4.4.2 音频播放器 (Player)	107	4.9.1 Activity 启动器 (ActivityStarter)	182
4.4.3 录音机 (SoundRecorder)	111	4.9.2 Web 客户端 (Web)	185
4.4.4 视频播放器 (VideoPlayer)	114	4.9.3 蓝牙客户端 (BluetoothClient)	193
4.4.5 摄像机 (Camcorder)	116	4.9.4 蓝牙服务器 (BluetoothServer)	195
4.4.6 照相机 (Camera)	117	4.10 乐高机器人® (LEGO MINDSTORMS)	201
4.4.7 图像选择框 (ImagePicker)	118	4.10.1 Nxt 指令发送器 (NxtDirectCommands)	202
4.4.8 文本语音转换器 (TextToSpeech)	121	4.10.2 Nxt 电机驱动器 (NxtDrive)	204
4.4.9 语音识别器 (SpeechRecognizer)	122	4.10.3 Nxt 颜色传感器 (NxtColorSensor)	205
4.4.10 Yandex 语言翻译器 (YandexTranslate)	124	4.10.4 Nxt 光线传感器 (NxtLightSensor)	207
4.5 绘图动画组件 (Drawing and Animation components)	126	4.10.5 Nxt 声音传感器 (NxtSoundSensor)	208
4.5.1 画布 (Canvas)	126	4.10.6 Nxt 接触传感器 (NxtTouchSensor)	209
4.5.2 图像精灵 (ImageSprite)	134		
4.5.3 球形精灵 (Ball)	140		
4.6 传感器组件 (Sensor components)	145		
4.6.1 计时器 (Clock)	146		
4.6.2 加速度传感器 (AccelerometerSensor)	152		
4.6.3 方向传感器 (OrientationSensor)	153		

4.10.7 Nxt 超声波传感器 (NxtUltrasonicSensor)	209	第 6 章 进阶项目	225
4.11 任意组件 (Any component)	210	6.1 涂鸦画板	225
4.12 项目：贪食球二	213	6.2 电子书——唐诗三百首	233
4.13 实验	217	6.3 跌倒求助	240
第 5 章 应用调试	218	6.4 天气预报	242
5.1 程序设计风格	218	6.5 智能题库管理	246
5.2 软件测试	219	6.6 用 App Inventor 直接控制乐高 EV3 机器人	267
5.3 调试应用	221	参考文献	272

第 1 章

App Inventor 入门

本章主要介绍 App Inventor 2 的发展、相关基础知识、开发环境搭建，以及界面构成等内容，是后续内容学习的基础。

1.1 App Inventor 简介

App Inventor 原是 Google 实验室 (Google Lab) 的一个子计划，由一群 Google 工程师和勇于挑战的 Google 使用者共同参与开发。App Inventor 于 2010 年 12 月 15 日被正式公开发布，团队由哈尔艾贝尔森 (Hal Abelson) 和马克弗里德曼 (Mark Friedman) 领导。在 2011 年下半年，谷歌发布源代码，终止其服务器，将该项目移交给麻省理工学院 (MIT) 移动学习中心，由 MIT 的 Hal Abelson 教授和他的同事埃里克克洛普弗 (Eric Klopfer) 和米歇尔雷斯尼克 (Mitchel Resnick) 教授领导继续开发。2012 年 3 月，MIT 版本的 App Inventor 被推出并开放使用。在 2013 年 12 月，MIT 发布了 App Inventor 2。

App Inventor 提供了一个完全在线开发的 Android 编程环境。通过它，使用者可以舍弃复杂的程式代码而使用积木块的堆叠法来完成 Android 程序。除此之外，它也正式支持乐高 NXT 机器人，这对于 Android 初学者或是机器人开发者来说是一大福音。

事实证明，基于可视“块”语言的编程方法，即便是对孩子来说，也是简单易用的。App Inventor 大大降低了为 Android 手机和设备开发应用的门槛。

MIT 版本的 App Inventor 正式发布半年后，在 2012 年 11 月 15 日-16 日举行的“2012 年 Google 中国教育高峰会”上，Hal Abelson 教授针对 App Inventor 进行专题演讲及介绍，与会的部分教师表现出了极大的兴趣。最终在 Google 的支持下，App Inventor 被迅速引入国内。2013 年上半年，国内举行了针对大学教师的师资培训班。在 2013 年和 2014 年，国内举行了多期针对中小学信息技术教师的师资培训班，以及成功组织了 2014 年 App Inventor 应用开发全国中学生挑战赛。越来越多的高校和职业技术学校将 App Inventor 作为课程，中学和小学将 App Inventor 作为信息技术课程教学内容或社团活动课等。

为配合在国内的推广，App Inventor 简体中文和繁体中文版于 2014 年 9 月被推出。随后，广州市教育信息中心搭建了国内的 App Inventor 服务器 (app.gzjkw.net)，并于 2015 年 1 月正式公开发布。

1.2 App Inventor 能做什么

这里借用大卫沃尔贝 (David Wolber) 和 Hal Abelson 出版的《App Inventor——创建你自己的

Android 应用》(*App Inventor-Create Your Own Android Apps*) 一书中的内容来回答这个问题。

1.2.1 玩

为你自己的手机创建应用是一件充满了乐趣的事, App Inventor 更促进了探索和发现的乐趣。只需在 Web 浏览器中打开 App Inventor, 连接上手机, 并把一些块拼在一起, 立即就能在手机上看到你建立的应用。例如: 你可以创建一个送给父母、老师和朋友的贺卡 App, 专属于你自己的日记本等。

1.2.2 建立原型

有了创建一个应用的想法, 然后该怎么办? 利用 App Inventor 可以快速地创建一个原型, 而不是将想法随手记在餐巾纸上, 或干脆让它随风飘散。原型是你不完整的和未加工的想法模型。用文字来表达一个想法, 就像写一篇散文给朋友或爱人; 而建一个 App Inventor 的原型, 就像写首诗歌给风险投资人。这样一来, 对于移动应用的开发来说, App Inventor 就像一张电子餐巾纸。

1.2.3 构建个性化应用

在当前的移动应用世界里, 我们被迫接受那些被推送过来的应用。谁没有抱怨过这些应用和期待个性化的应用, 或者可以以某种方式调整它的功能。使用 App Inventor, 可以构建如你自己所想的应用。

1.2.4 开发完整的应用

App Inventor 不只是一个原型系统或界面设计器, 也可以用于创建完整的、多用途的应用。它所使用的语言提供了所有基础的编程指令, 如循环和条件语句等, 只是以“块”的方式来呈现。

1.2.5 教与学

无论你是在初中、高中, 还是在大学, App Inventor 都是一个伟大的教学工具。不仅仅是针对计算机科学, 即使是对数学、物理以及其他学科, 甚至创业来说, 它都是一个了不起的工具。通过它, 可以实现在创造中学习, 而不是死记公式。

1.3 App Inventor 开发环境搭建

1.3.1 系统要求

1. 操作系统

- ☐ Macintosh (使用 Intel 处理器): Mac OS X 10.5 或更高版本。
- ☐ Windows: Windows XP, Windows Vista, Windows 7 或更高版本。
- ☐ GNU/Linux: Ubuntu 8 或更高版本, Debian 5 或更高版本。

注: GNU/Linux live 环境下开发, 在计算机和 Android 设备之间仅支持 Wi-Fi 连接。

2. 浏览器

- ☐ Mozilla Firefox 3.6 或更高版本。

- ❑ Apple Safari 5.0 或更高版本。
 - ❑ Google Chrome 4.0 或更高版本。
- 不支持 Microsoft Internet Explorer。

3. 模拟器

- ❑ Phone、Tablet 或 Emulator（模拟器）。
- ❑ Android Operating System 2.3 或更高版本。

1.3.2 服务器地址

App Inventor 2（本书后面均简称为 AI2）的官方服务器地址为：<http://ai2.appinventor.mit.edu>。使用官方服务器的地址进行开发时，需要使用 Google 的 Gmail 邮箱账号和密码作为登录账号和密码，并由 Gmail 服务器进行账户的认证，因此，从中国国内访问官方服务器可能存在问题。建议大家访问国内的服务器：<http://app.gzjkw.net>。

1.3.3 三种测试应用的方法

AI2 是完全基于浏览器开发安卓应用。在开发过程中，根据选用的开发方式不同，用户可能需要在本地计算机配置和安装开发环境。

用 AI2 构建应用时，有三种可选择的方式设置实时测试（可参考 <http://appinventor.mit.edu/explore/ai2/setup.html>）。

1. 使用安卓设备和无线网络进行开发（官方推荐开发方式）

通过这种方式，用户不需要下载任何软件到计算机，可直接在服务器上进行开发。为了在安卓设备上实时测试，用户需要安装“MIT App Inventor Companion App”到安卓设备上。安装完成“Companion”之后，就可以在浏览器中打开项目，在设备上运行“Companion”，然后就可以测试构建的应用程序，如图 1.1 所示。具体步骤如下。

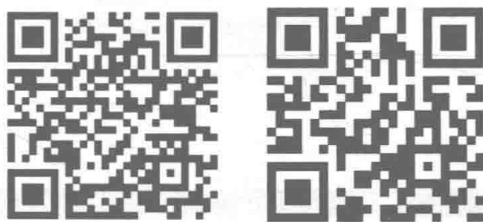


(a) 在计算机上创建项目 (b) 在安卓设备上实时测试

图 1.1 使用安卓设备和无线网络开发

步骤 1：下载和安装“MIT App Inventor Companion App”。

可以通过扫描“Google Play Store”[图 1.2 (a)]或“Apk 文件”[图 1.2 (b)]二维码进行下载安装。



(a) Play Store (b) Apk 文件

图 1.2 MIT App Inventor Companion 下载地址

此外，也可以直接通过访问地址 <http://appinv.us/xAI2cf34> 下载 Apk 文件，然后安装到安卓设备。

注：（1）如果你的安卓设备不能使用二维码扫描，也可以直接到 Google Play Store 搜索“MIT AI2 Companion”，然后下载安装。

（2）如果直接使用 Apk 安装，请将安卓设备设置为允许安装“未知来源”，具体方法为：Android 4.0 以前版本，可通过“设置”→“应用”→“未知来源”进行设置；Android 4.0 或之后版本，可通过“设置”→“安全”→“未知来源”进行设置。

步骤 2：将计算机和安卓设备连接到同一无线网络。

步骤 3：登录服务器。

打开网站 <http://app.gzjkw.net>，如图 1.3 所示。

如果是第一次登录，请单击“申请新账号/重设密码”（或单击用“QQ 账号登录”），将打开申请注册新账号网页，如图 1.4 所示。



图 1.3 登录界面

申请注册新账号，或者要求重设密码链接

你可以设置你账号的初设密码；如果你忘记了你的密码，你可以申请改变你的旧密码。

输入你的电子邮箱地址：

[发送链接](#)

图 1.4 申请注册新账号，或者要求重设密码链接

在“输入你的电子邮箱地址：”文本框输入常用邮箱，然后单击“发送链接”按钮。

然后到输入的邮箱里面找到由 appinventor@gzedu.gov.cn 发出的邮件。在邮件正文中找到“你的链接是：”后面的链接，打开该链接，进入到设置密码网页，输入准备设置的密码，然后单击“设置密码”按钮完成注册。系统会出现“广州市教育信息中心 App Inventor 网站用户使用协议”条款，单击最后的“我接受以上服务条款”按钮将自动登录到 AI2 开发界面，如图 1.5 所示。

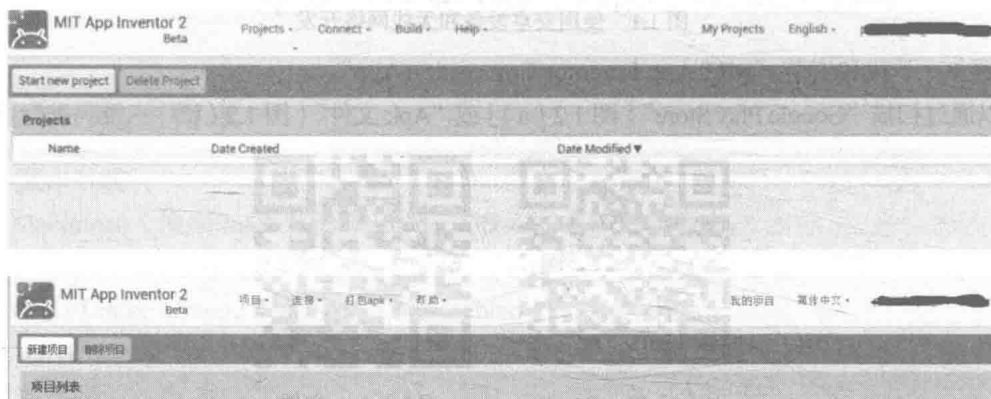


图 1.5 AI2 初始中英文界面

步骤 4: 创建 App Inventor 项目，并连接到安卓设备。

在图 1.5 所示的界面中单击“Start new project (新建项目)”，打开“Create new App Inventor project (新建项目……)”对话框，如图 1.6 所示。输入项目名称，单击“OK”按钮完成项目创建。可以任意选择一个组件放在工作面板中。



图 1.6 “新建项目”的中英文界面



项目名称必须以字母开头，且只能由字母、数字或下划线组成，名称中间不能包含空格，且区分大小写，如“Hello”和“hello”是两个不同的项目名。

接下来在图 1.7 的“Connect (连接)”中选择“AI Companion (AI 伴侣)”。

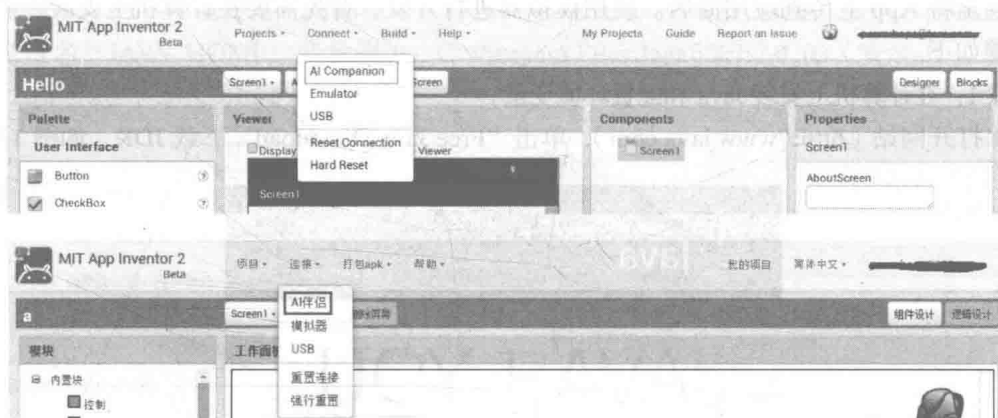


图 1.7 连接“AI Companion”

然后弹出扫描二维码对话框，如图 1.8 所示。在安卓设备中启动安装好的“MIT App Companion”应用。然后单击“Scan QR code”按钮扫描浏览器中的二维码。

几秒钟后，构建的 App 就会显示在安卓设备上。如果用户在“Designer (组件设计)”或“Blocks (逻辑设计)”进行了更改，安卓设备会及时更新 App。

问题排除：

如果用户的 App 没有出现在安卓设备上，可能的原因如下：

- 安卓设备上配套安装的 App Inventor Companion App 版本过旧，请从 AI2 网站下载最新的 Companion App。



图 1.8 扫描二维码

- ❑ 安卓设备没有连接到无线网络，请确认安卓设备上的 AI Companion App 屏幕下方显示了 IP 地址。
- ❑ 安卓设备和计算机没有连接到同一无线网络。
- ❑ 用户所在的学校或组织可能不允许使用无线网络连接协议。

2. 没有安卓设备，安装并运行 AI2 模拟器

如果没有安卓手机或平板电脑，用户仍然可用 AI2 调试 App。AI2 提供了一个安卓模拟器，就像安卓设备一样，但可以运行在电脑屏幕上。所以可以通过模拟器测试 App 并将 App 分发给其他人，甚至将 App 上传到应用商店。使用模拟器进行开发，首先需要在计算机上安装一些软件，具体步骤如下。

步骤 1：在计算机上安装 Java 和配置环境变量。

(1) 打开网站 (<http://www.java.com>)，单击“Free Java Download”下载 JDK，如图 1.9 所示。



图 1.9 下载 Java

然后选择“Agree and Start Free Download（同意并开始免费下载）”。下载完成后，双击下载的文件进行安装。安装完成后，进行环境变量的设置。下面以 Windows 7 为例讲解设置环境变量的方法。

(2) 用鼠标右键单击“计算机”->“属性”->“高级系统设置”，如图 1.10 所示。



图 1.10 计算机属性

(3) 打开“系统属性”对话框后，单击“高级”→“环境变量”，如图 1.11 所示。

(4) 在打开的“环境变量”对话框中，单击“系统变量”下的“新建”按钮，添加以下环境变量，变量名：JAVA_HOME，变量值为：C:\Program Files\Java\jdk1.7.0_03（安装 JDK 的目录），如图 1.12 所示。

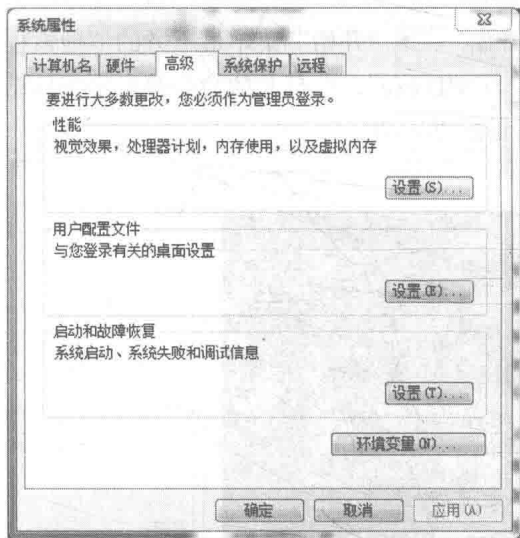


图 1.11 系统属性

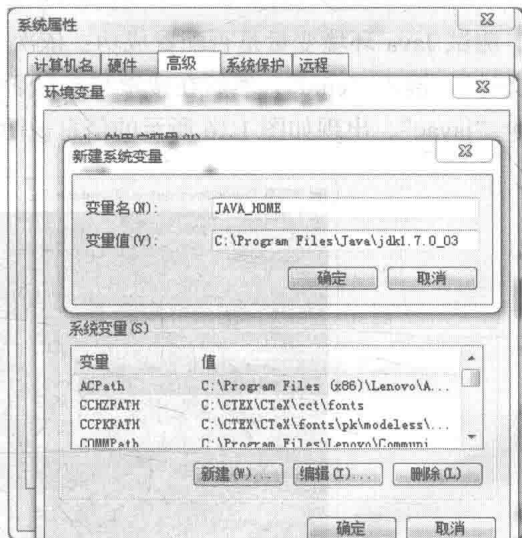


图 1.12 新建系统变量

(5) 在系统变量中选择“Path”项并单击“编辑”，在打开的“编辑系统变量”对话框中的“变量值”文本框末尾添加“;%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin”。

原有的路径要保留，前面用分号隔开，如图 1.13 所示。

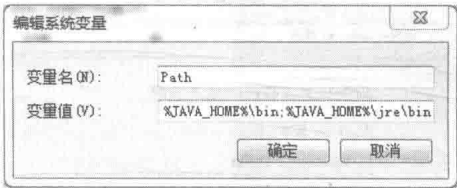


图 1.13 编辑系统变量

(6) 最后单击“系统变量”下的“新建”按钮，添加以下环境变量，变量名：CLASSPATH，变量值为：.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar（要加“.”表示当前路径），如图 1.14 所示。

设置环境变量中的步骤（4）～（6）可以简化，用下面的设置即可。在系统变量中选择“Path”项并单击“编辑”，在打开的编辑系统变量对话框中的“变量值”文本框末尾添加“C:\Program Files\Java\jdk1.7.0_03\bin”。

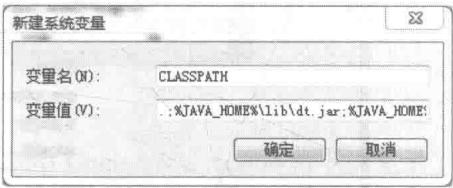


图 1.14 新建系统变量

原有的路径要保留，前面用分号隔开，如图 1.15 所示。

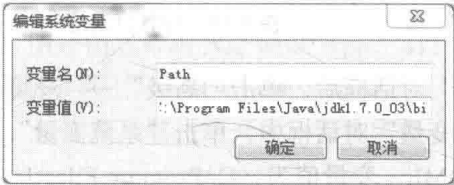


图 1.15 编辑系统变量

测试 Java 环境变量是否配置成功：依次单击“开始” -> “开始”菜单最下方的“搜索程序和文件”，键入“cmd”，然后在搜索结果对话框选择“CMD.exe”，在打开的命令符提示窗口键入命令“javac”，出现如图 1.16 所示的信息说明环境变量配置成功。

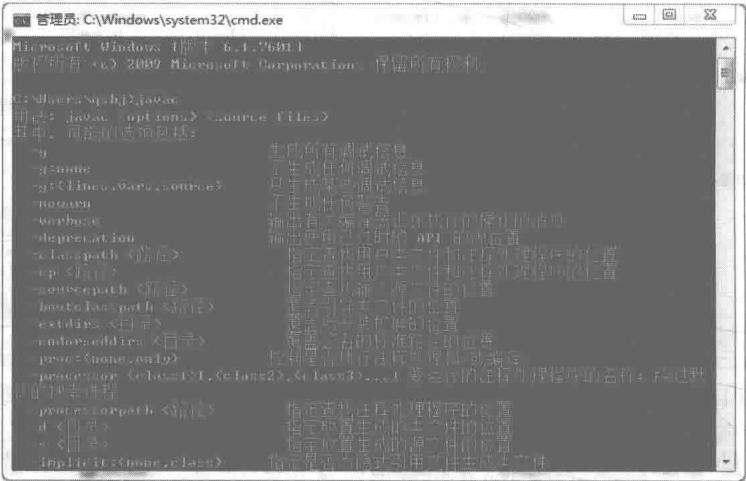


图 1.16 测试环境变量是否配置成功

步骤 2：在计算机上安装“App Inventor 2 Setup”软件包。

如果已经安装了之前版本的“App Inventor 2 Setup”安装包，在安装最新版本之前需要卸载它们。

下载软件包（http://appinv.us/aisetup_windows）后，要以 Windows 7 的管理员的权限进行安装。

步骤 3：启动“aiStarter”。

安装完成后，在桌面或“开始”→“所有程序”中启动“aiStarter”。启动后会显示图 1.17 所示窗口，请不要关闭该窗口，可以将其最小化到任务栏。

步骤 4：打开 App Inventor 项目和连接到模拟器。

注册账号、登录网站和创建项目的方法可参考方式 1 中的介绍。由于 AI2 已经提供了中文版本，因此在后面主要以中文版本进行介绍，关键地方进行中英文对照讲解。要在不同语言版本间进行切换，可单击 AI2 顶部的语言图标，然后在弹出的菜单中选择相应语言版本即可，如图 1.18 所示。

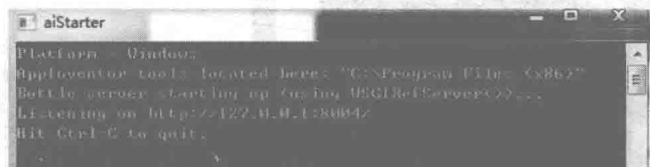


图 1.17 “aiStarter”启动界面

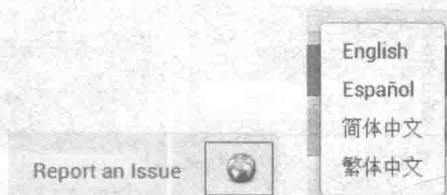


图 1.18 选择语言

项目建好后，从“连接（Connect）”中选择“模拟器（Emulator）”，可启动模拟器，如图 1.19 所示。

浏览器会弹出一个“连接中……（Connecting……）”对话框，如图 1.20 所示。

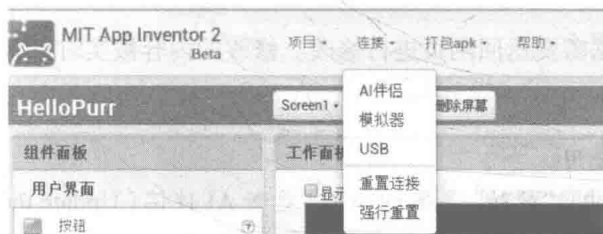


图 1.19 “连接”菜单

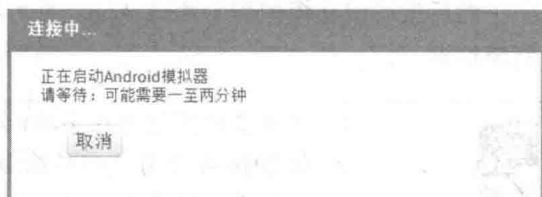


图 1.20 “连接中”对话框

在启动过程中，系统会检查模拟器中的伴侣程序是否过期。如果不是最新的版本，系统会提示“伴侣程序已过期，……”，如图 1.21 所示，选择“确定（OK）”按钮可进行更新。

系统接着会弹出“软件升级”界面（图 1.22）和“升级完成”提示界面（图 1.23）。

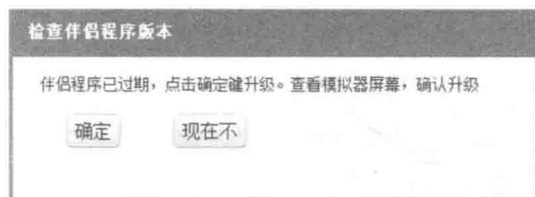


图 1.21 检查伴侣程序版本

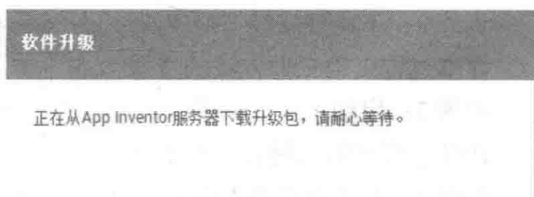


图 1.22 软件升级

出现图 1.23 的界面后，把屏幕切换到模拟器界面，如图 1.24 左图所示。选择“确定（OK）”