第一章PencilCode环境介绍

本节内容主要向大家介绍PencilCode的编辑环境。PencilCode是由谷歌工程师DavidBau和他的儿子AnthonyBau一起开发的应用于多种领域的开源技术，只需要登陆PencilCode.net网站就可以免费使用该工具。

两种观看模式

PencilCode提供两种视窗界面。电脑用户通过输出设备就可以进行观看。一个项目的运行合计离不开项目源代码，在PencilCode中，屏幕被分成了两部分，左边的一部分用于显示项目的源码，而右边的一部分则输出代码的运行结果。用户可以通过点击屏幕中间的“play”键来运行已经编辑好的源代码。如图1 所示。



PencilCode提供的语言和函数库

PencilCode在块和文本编辑过程中支持主流的web编辑语言，常用的函数库包括：

HTML,web标准语言

JavaScrip，web浏览器标准的项目语言

CSS，

通过PencilCode，学员可以使用这些工具创建各种全球可见的web应用，并且探索过程中基本的计算思维概念产生影响。

我们将在本书后续章节中介绍这些功能的使用。

A Web Page is a Program

每一个网页都是一个项目，都具有源代码和显示两部分。

当用户点击链接打开网页的时候，网页源代码就被发送到了用户主机。页面可以是不同语言构成，例如HTML和JavaScrip。但是显示结果都是由浏览器编译过的界面。例如图2和图3所示。





鼓励学生可以去[www.ebay.com，www.flicker.com，www.mozilla.org](http://www.ebay.com，www.flicker.com，www.mozilla.org)之类的网站，通过查阅网页源码中的隐藏信息探索不同网站的源代码。借助Ctrl-U查阅源码。

基于网页的PencilCode

PencilCode的项目编辑都是基于网页进行的。顶端编辑菜单仅提供几个用于控制页面共享的编辑按钮。右上角的按钮用于项目的保存、共享页面管理以及帮助文档。左上角显示项目名称。项目重命名通过点击褐色名称进行编辑。如图4所示。



项目名称用于设置项目网络链接地址（URL web address），如下表所示：

|  |  |  |
| --- | --- | --- |
| 用户名 | 项目名称 | 网站链接 |
| Coolsite | First | http：//coolsite.pencilcode.net/home/first |
| David | Example/posterize | http://david.pencilcode.net/home/ example/posterize |

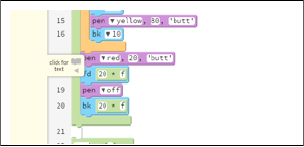
鼠标悬停在蓝色区域右边的“output”按钮上会激活“open window”链接，打开一个新的页面用于显示已经完成的项目成果。这个页面不会显示任何代码，（这个链接仅对已经记录和保存的项目有效。）这个操作用于全屏运行编辑的项目结果，使用Ctrl-U打开“view source”查阅页面代码。

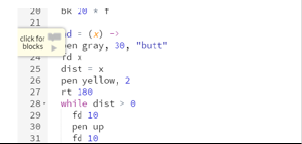


在PencilCode，全屏输出的URL地址含有“/home/”字段，因此这个地址可以被链接、邮件发送、植入到任何地方。将“/home/”字段改为“/edit/”的URL地址显示进入UI编辑，显示PencilCode项目源码。

什么样的程序设计语言？

PencilCode支持用户使用“模块化”语言，通过对编辑模块的拖拽来编辑设计程序。通过这些已编辑的模块可以直接调用文本语言诸如CoffeeScript、JavaScrip及HTML。尽管这些模块由不同的文本代码构成，但是都属于以可视化的方式编辑程序的语言。



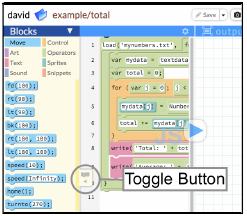


程序设计语言与其他语言一样，既能支持硬件设备也能被程序员理解。英文“一遍又一遍的调试”在英特尔奔腾芯片中被解释为“1110101111111110”。英文单词对于计算机而言太模糊，而机器代码对于程序员又太过于晦涩。程序设计语言使用可读单词，具有精确的语法编写规则，使得计算机能够精确地读写。

当阅读CoffeeScript或者JavaScrip的可视化模块时，编程语法以块的拼接结构形式表现出来。例如不同命令下的编辑语言，以不同形态的模块显示出来。对于一个控制语句中的命令，则以命令模块的嵌套来显示。

模块和文本的转换

在PencilCode中，块编辑和文本编辑的便捷都被展示出来。块编辑将CoffeeScript、JavaScrip或者HTML语言的语法以可视化的形式展示出来，方便学生在选取和操作文本和指令块。



黄色标签页左下角的开关用于从命令块中查找适合进行块编辑的文本，通过点击块或者点击文本来进行选取。

何时使用块和文本？

当学生学习一个新的函数或者命令的时候就可以学习使用块编辑。块根据不同的种类被设置了不同的颜色，使用时需要根据缺口进行正确的拼接。用户只需合理的设计块的凭借，因此每一个新想法都能很容易的实现。

当学生足够了解函数和命令后，为帮助他们记忆可以使用块编辑。当学生熟悉了一部分项目语言后，他们会发觉键盘录入会比拖动块编辑更加快速和流畅。

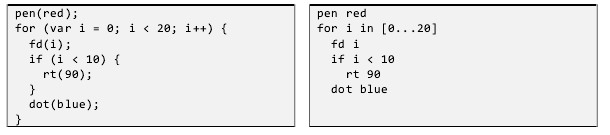
在PencilCode中，基于代码的块包含了编辑语言中基本的数学应用。例如，当使用CoffeeScript语言，使乌龟向前移动100像素读取为“fd 100”，这个相同的代码被写入文本模式。当学生使用文本编辑模式修改或者添加代码时，他们可以回到块编辑页面查阅块的变化。

学生无论使用块编辑还是文本编辑都会感觉很简单。最初我们要求学生只使用主要的块，他们会像录入文本那样录入代码。更多的学生会很自然的将块和文本进行结合在他们所熟悉的编辑语言中。

CoffeeScript的开始

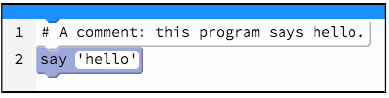
PencilCode同时支持CoffeeScript和JavaScript，但是默认的语言是CoffeeScript，因此我们推荐学生学习CoffeeScript。

CoffeeScript应用于在很多的技术中心，包括Github和Dropbox。它的运行速度和功能与JavaScript相同。CoffeeScript，语法更简单，使得学生很容易做出第一个程序。也阐明了代码中的一些概念，诸如函数、循环、嵌套、录入和数组。



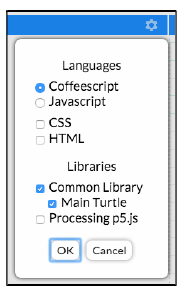
命令

命令标记：在块编辑中创建命令，第一个创建的块通过“Enter”键并且在块的头部录入#号。块命令如下图所示：



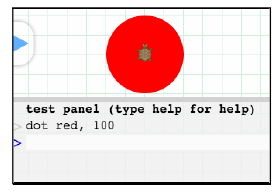
语言选择：

在CoffeeScript和JavaScript中进行选择，点击蓝色标题中的“gear cion”。从中选择语言和添加的函数库，窗格语言可以选择CSS或者HTML。

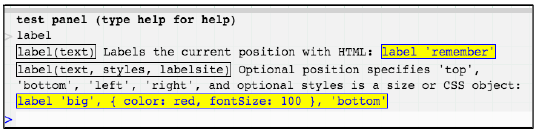


测试区域

位于页面右下方的测试栏可以帮助核对CoffeeScript和JavaScript的语法。例如，录入“dot red，100”点击运行，可以看到命令的执行状况和功能。

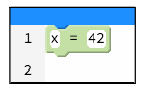


在测试区域录入没有参数的命令，基本得不到什么帮助。如果帮助信息显示不下，深灰色的区域会延长用于显示帮助文档。



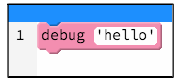
测试框调试

可以通过测试框调试项目变量。例如，运行左侧区域的指令“x=42”，



录入在测试框中看到的“x”的值，并点击回车键。如果测试框显示“x is not defined”，意味着程序还没有开始运行，需要点击“play”按钮，之后会看到程序的运行结果“x”的值是42。

“debug”命令很特殊，可以直接在主页面上输出。尝试创建项目，设置“debug ‘hello’”，单词“hello”被显示出来。



运行的时候，测试框会显示“hello”。

PencilCode的库

PencilCode提供哪些可用的函数库？大约100个函数存在于PencilCode的块列表中，同时PencilCode还提供一个大型开放式的函数库远超出列表中提供的函数数量。基本上所有的页面都可以用。

在教师手册中只对其中的一小部分函数进行了讨论说明，而剩余的库可以通过互联网上的例子和教程进行学习使用。PencilCode可用库包括：

1.网络文档对象模型（DOM）。互联网标准委员会制定，该模型函数可以被互联网上所有页面识别。

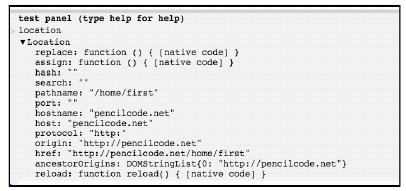
2.jQuery.最受欢迎的互联网页面库，应用于大多数网站。我们将在第11章介绍jQuery库的使用方法。

3. jQuery-turtle。PencilCode提供的乌龟库拓展了jQuery库。提供了本手册第一部分介绍的简单易用的功能。大多数函数都可以通过乌龟库进行块编辑。

4.socket io.实时通信库可以实现与浏览器的即时通信。

乌龟库之外的巨大功能库讲解

尽管网页编辑有不同的功能，所有的标准类都可以被测试窗口解析。例如，录入“location”到DOM界面的“location”对象，并且通过点击鼠标使其扩大。测试窗口显示“location”包含血多函数和大量的数据，“href”：一个项目使用location.href变量。



如何使用本书

这本教师手册用于帮助学生学习基本的项目设计知识。好像有教师在课程中做项目介绍一般。这本手册告诉学生如何分阶段的使用PencilCode环境，以及开始编写简单的程序。每章围绕着基本的程序结构，从基本的概念到深入的应用。手册也告诉学生如何讲块代码编译为代码文本：开始编程可能会让学生望而生畏，最初使用块编辑可以克服恐惧的心理。同时关注学习程序设计方法，以及程序所要解决的问题。

教师手册基于CSTAK-12标准编写。每章都按照教案的标准框架建立。各章节的基础项目由David和Deepa完成，更多有效应用参阅网站向导。

如何阅读这本书

这本手册适用于借助提供一些PencilCode教授程序设计的教师。每章提供的截图用于帮助教师完成每个主题。关键片段可以给老师一个相关技术主题的概述。每章都会提供一个关键词，然而教师的课程设计向导覆盖每个程序。