

编程起步

欢迎走进奇妙的 Scratch 编程世界，从这里开始，本书将向没有编程基础的读者讲授如何使用 Scratch 编写应用程序。这一章将向初学者讲授学习本书其余部分内容需要掌握的一些基本概念和基础知识。

什么是编程？什么是 Scratch 编程？Scratch 积木式编程具有哪些优势？Scratch 编程语言的指令系统是怎样的？读者将在阅读本章的过程中找到这些问题的答案。同时，通过对本章的学习，读者将学会如何选择 Scratch 开发环境，学会安装 Scratch 离线编辑器和设置语言环境。之后，本书将手把手地教读者使用 Scratch 编辑器开发自己的应用程序项目，通过简单地临摹案例让读者快速熟悉 Scratch 开发环境和编程方法，为后续的学习作铺垫。此外，如果读者打算以后学习某种高级语言（如 Python、C/C++ 等），可参考本章给出的从 Scratch 到 Python 等高级语言的学习路径的建议进行学习。

本章包括以下主要内容。

- 介绍 Scratch 编程的特点、主要版本和项目构成等。
- 准备 Scratch 开发环境和设置编辑器的语言环境。
- 介绍 Scratch 编辑器界面的主要组成部分及其功能说明。
- 介绍 Scratch 指令系统和指令积木的特点、操作方法、功能类别等。
- 以临摹方式创作“韩信点兵”和“星际飞行”项目。

1.1 Scratch 编程概述

1.1.1 为什么用 Scratch 编程

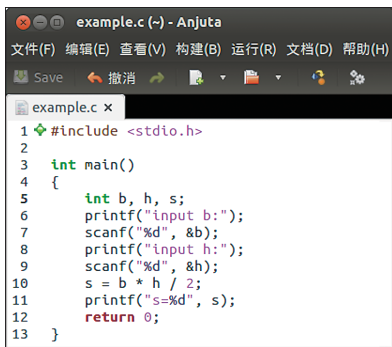
在世界上第一台电子计算机 ENIAC 诞生后，各种编程语言陆续被计算机科学家创造出来。人们通过编程语言能够高效地与计算机系统进行交流，控制计算机按照人们的意愿进行工作。经过不断地发展和完善，一些编程语言与人类的自然语言和数学语言越来越接近，它们被称为高级语言，比如 C、C++、Java 和 Python 等就是全世界较为流行的高级语言。

为了利用计算机进行工作，人们使用某种编程语言将解决问题的方法和步骤描述成计算机能够理解和执行的一系列指令，这些指令的集合叫作计算机程序（简称程序，也叫作脚本），这个过程叫作编写程序（简称编程）。

如图 1-1-1 所示，这是一个使用 C 语言编写的计算三角形面积的程序。从中可以看到，C 语言使用英文字符来描述程序的指令（也叫作代码），其他高级语言也是如此。这是由

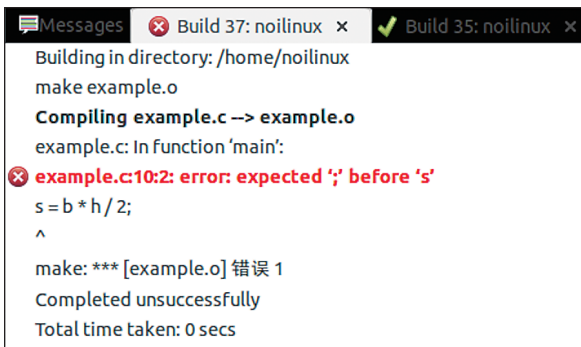
于美国在早期计算机工业的发展中处于主要地位，因此一般的高级语言都是以英语为蓝本进行设计的。

我们还看到在这个 C 语言程序中，每一行代码都以一个分号结束。如果某一行代码末尾缺少一个分号，那么这个程序在编译时就会出现语法错误。如图 1-1-2 所示，这是在某个 IDE（集成开发环境）中编译时因缺少分号而提示的错误信息。



```
example.c (-) - Anjuta
文件(F) 编辑(E) 查看(V) 构建(B) 运行(R) 文档(D) 帮助(H)
Save 撤消 取消 运行 编译 构建 运行 文档 帮助
example.c x
1 #include <stdio.h>
2
3 int main()
4 {
5     int b, h, s;
6     printf("input b:");
7     scanf("%d", &b);
8     printf("input h:");
9     scanf("%d", &h);
10    s = b * h / 2;
11    printf("s=%d", s);
12    return 0;
13 }
```

图 1-1-1 计算三角形面积的 C 语言程序



```
Messages Build 37: noilinux x Build 35: noilinux x
Building in directory: /home/noilinux
make example.o
Compiling example.c -> example.o
example.c: In function 'main':
x example.c:10:2: error: expected ';' before 's'
s = b * h / 2;
^
make: *** [example.o] 错误 1
Completed unsuccessfully
Total time taken: 0 secs
```

图 1-1-2 编译 C 语言程序时缺少分号出现的错误信息

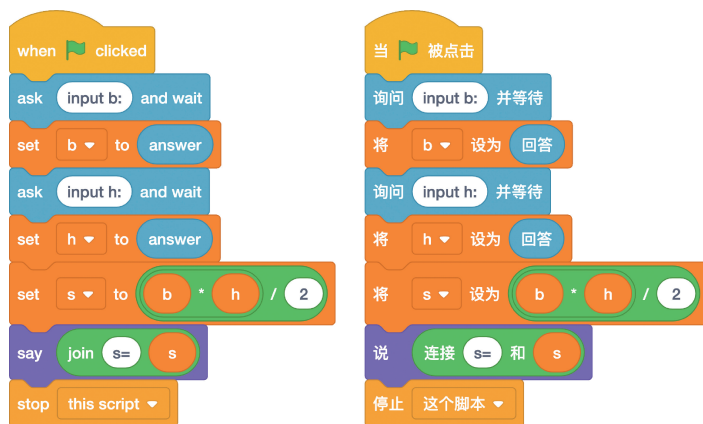
和 C 语言类似，其他高级语言也都有着严格的语法要求和各种编程规则。如果我们编写的程序没有遵守这些“金科玉律”，那么，轻则收到错误或警告信息，重则可能导致计算机系统宕机。因此，在实际工作中进行编程需要经过专门的学习和训练，由专业程序员负责。

编程爱好者在学习 C 语言等编程语言时，一开始总是会遇到各种各样的麻烦和困难。尽管在计算机发展史上，很早就出现了诸如 BASIC 这种专门给普通编程爱好者使用的编程语言，并且在青少年中也有一定的使用量，但是这类编程语言仍然有着严格的语法要求和编程规则，将许多初学者的编程热情浇灭在起步阶段。

近年来，随着全球信息技术的飞速发展，专门为青少年开发的各种高级语言不断涌现，其中以 MIT Scratch 为代表的图形化编程语言从中脱颖而出，逐渐成为全球流行的、最适合对青少年进行编程教育的新一代编程语言。

如图 1-1-3 所示，这是使用 MIT Scratch 软件编写的计算三角形面积的两个程序代码，它们分别是使用英文和中文进行描述的。这种使用图形化编程语言编写的程序，由代表不同指令的积木块按照一定的逻辑关系组合而成。不需要背记复杂的语法规则，也不用担心程序会出现语法错误而无法运行，或者因为违反某些编程规则而导致程序崩溃。如此一来，就算是初学者也能将精力放在思考程序逻辑上，能够轻松地按照自己的意图选择不同功能的积木块，再将它们拼接组合成自己需要的程序。这种编写程序的方式充满了玩乐高积木般的乐趣。因此，Scratch 在 2007 年一经推出，就像一团熊熊烈火，迅速点燃了全球青少年的编程热情。

本书正是要向读者讲授图形化编程语言——Scratch，它是由美国麻省理工学院（MIT）媒体实验室专门为 8~16 岁青少年设计开发的。使用这个具有魔力的 Scratch 作为编程工具，能够轻松地创作出各种交互式故事、游戏、动画、音乐、美术作品或其他应用程序，并通过网络社区将自己的创意作品分享给全世界的编程爱好者。



(a) 英文代码

(b) 中文代码

图 1-1-3 两个计算三角形面积的 Scratch 程序

1.1.2 从 Scratch 到 Python

近年来，在中小学阶段推广和普及编程教育已成为全球各国的共识。2017年7月，国务院发文指出，将逐步在中小学阶段设置人工智能相关课程和推广编程教育。在全球流行编程教育的趋势之下，作为适合中小学生学习编程入门教育的图形化编程语言，Scratch 的影响力日渐扩大，学习 Scratch 编程的青少年与日俱增。

Scratch 是青少年编程教育的起点，而不是终点。为使 Scratch 能够被 8~16 岁的青少年学习和理解，Scratch 开发团队刻意限制 Scratch 编程语言的功能和特性，保持 Scratch 简单易学的特色。因为 Scratch 的设计初衷是帮助青少年学习编程，而不是进行专业软件开发。

编程思想是灵魂，编程语言只是躯壳。当青少年通过 Scratch 掌握基本的编程思想之后，可以选择转向 Python 等具备完整编程特性的高级语言，就可以开发更为复杂的网络应用程序、数据库应用程序或人工智能应用程序，等等。

那么，从 Scratch 到 Python 的学习之路应该如何前进呢？下面给出一个学习路径的建议供读者参考。

第 1 步：在中文界面下学习 Scratch 编程。

在国内，8 岁的儿童正处于小学低年级阶段，很多地方是从小学三年级起开设英语课。无论是汉字或英文，低龄儿童都存在认知上的困难。Scratch 图形化的特性有助于少年儿童学习编程。对多数人而言，在英文界面下进行 Scratch 编程有一定的困难，而 Scratch 对简体中文或其他众多语言的支持能把编程变得简单。如图 1-1-3 (b) 所示的程序是在简体中文界面下编写的，Scratch 指令积木块上的文字采用简体中文表示，这样更容易阅读和理解。同时，也可以减少为 Scratch 程序添加注释。因此，推荐初学者先在简体中文界面下学习 Scratch 编程。

第 2 步：在英文界面下学习 Scratch 编程。

在简体中文界面下学习并掌握 Scratch 编程之后，就可以切换到英文界面下进行编程。

如果初学者有一定的英文基础,完全可以跳过第 1 步,直接使用全英文进行编程。

在英文界面下进行 Scratch 编程,是为以后学习 Python 等高级语言做准备。这是因为 Python 等高级语言编程是基于文本的、使用英文关键字编写程序代码的。所以,初学者在中文界面下掌握 Scratch 编程之后,作为一个过渡阶段,需要切换到英文界面去适应在英文环境中进行编程。

如图 1-1-4 所示,这是在 Scratch 英文界面下编写的插入排序算法的程序代码。我们把它积木块图形去除只保留文字,再把它和使用 Python 语言编写的插入排序算法的代码进行对比,如图 1-1-5 所示,从文本的角度进行对比,可以看到 Scratch 和 Python 的程序是很相似的。因此,初学者先学习 Scratch 编程,在掌握基本的编程思想之后,再转向学习 Python 等高级语言编程,学习曲线会比较平缓。

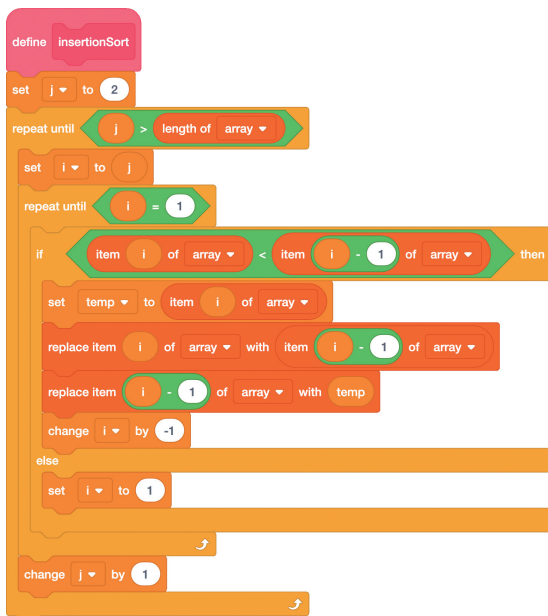


图 1-1-4 用 Scratch 编写的插入排序算法

```

1  define insertionSort
2  set j to 2
3  repeat until j > length of array
4  set i to j
5  repeat until i = 1
6  if item[i] of array < item[i-1] of array then
7  set temp to item[i] of array
8  replace item[i] of array with (item[i-1] of array)
9  replace item[i-1] of array with (temp)
10 change i by -1
11 else
12 set i to 1
13 change j by 1

```

(a) Scratch 代码

```

1  def insertionSort():
2  j = 1
3  while j < len(array):
4  i = j
5  while i > 0:
6  if array[i] < array[i-1]:
7  temp = array[i]
8  array[i] = array[i-1]
9  array[i-1] = temp
10 i = i - 1
11 else:
12 i = 0
13 j = j + 1

```

(b) Python 代码

图 1-1-5 使用 Scratch 和 Python 编写的插入排序算法

第 3 步: 学习 Python 或其他高级语言编程。

在通过 Scratch 走上编程之路后,可以继续学习 Python、C/C++、Java 等高级语言。

在众多的高级语言中,Python 是一个不错的选择。它有着庞大的社区支持,各种技术资料非常丰富。同时,它有众多的编程库能够实现对各种新技术的支持,比如当下火热的人工智能和机器学习领域,都能找到相应的 Python 库。无论是客户端、云端,还是物联网终端,都能看到 Python 的身影,可以说,Python 的应用无处不在。总之,Python 的优点很多,在此不一一列举。

对 Python 或者其他高级语言编程知识的讲授,已经超出本书的范畴,读者可以购买相关图书进行学习。推荐图书:《Python 趣味编程:从入门到人工智能》,ISBN 978-7-302-52820-3,清华大学出版社。

1.1.3 Scratch 的主要版本

2007年5月，Scratch 软件的第一个正式版本发布，它基于 Squeak 平台和 Smalltalk 语言开发，可以运行在 Windows、macOS X 和 Debian/Ubuntu 等操作系统上。2009年，Scratch 1.4 版本发布，其界面外观如图 1-1-6 所示。作为第一代 Scratch 软件的最后一个版本，目前仍然有少量用户在使用。如果你的计算机硬件配置较低，或者使用的是 Windows XP 操作系统，则可以安装使用 Scratch 1.4 编辑器。



图 1-1-6 用 Scratch 1.4 编辑器创作“鲨鱼吃小鱼”游戏项目

2013年5月，具有里程碑意义的 Scratch 2.0 正式版发布，其界面外观如图 1-1-7 所示。作为第二代 Scratch 软件，它是基于 Adobe Flash 技术全新设计和开发的，分为在线版和离



图 1-1-7 用 Scratch 2 离线编辑器创作“雷电战机”游戏项目

线版两种。在支持 Adobe Flash Player 插件的浏览器（IE、Edge、Firefox、Safari 或 Chrome 等）中可以直接运行 Scratch 2 在线编辑器，不需要安装到用户的计算机上，极大地方便了用户在线进行编程；而在安装有 Adobe AIR 运行环境的操作系统（Windows、macOS X 或 Linux）中，可以运行 Scratch 2 离线编辑器。

和 Scratch 1.4 相比，Scratch 2 可谓是焕然一新，它新增的功能主要有：提供更为友好的图形用户界面；升级内置的声音编辑器和图像编辑器；通过支持矢量图形而改善舞台画面的质量；在线版提供的书包功能可以方便地管理常用的图片、声音和脚本等资源；增加视频侦测功能，可实现手势控制等简单的体感技术应用。最为重要的是增加了克隆功能和自定义过程功能，这能极大简化复杂应用程序的编写，提高代码的复用程度，使在 Scratch 2 中能够学习和应用面向对象和模块化的编程思想，有利于以后转向其他高级语言的学习。

随着 Flash 技术逐渐被淘汰，MIT Scratch 官方团队转而使用当前流行的 HTML5 技术重新设计和开发第三代 Scratch 软件。

2019 年 1 月，Scratch 3.0 正式版发布，其界面外观如图 1-1-8 所示。Scratch 3 新设计的界面布局与第一代 Scratch 相似，指令积木的配色比 Scratch 2 更加赏心悦目，积木文字的显示更加清晰。在用户操作上，Scratch 3 增强了对鼠标操作的支持，用户可以非常便捷地选取指令积木、缩放和拖动代码，编程的体验更加友好。在性能上，Scratch 3 提供了更好的图形渲染引擎，代码运行效率更高。



图 1-1-8 用 Scratch 3 离线编辑器创作“抢滩登陆”游戏项目

Scratch 3 并没有带来革命性的变化，而是秉持简单易用的设计原则，向下兼容 Scratch 2，所有使用 Scratch 2 创作的项目都可以在 Scratch 3 中正常运行。对于习惯 Scratch 2 的用户来说，只要稍加适应即可流畅地使用 Scratch 3 进行编程。

1.1.4 Scratch 项目概述

一般来说，Scratch 编程指的是使用 Scratch 编辑器创作交互式故事、动画、游戏、音乐、

美术作品等各种类型的项目（作品）。

一个 Scratch 项目（Project），通常由舞台（Stage）、角色（Sprite）、代码（Code）和声音（Sound）等基本要素构成（见图 1-1-9），它就像是在剧场的舞台上表演的一出话剧（项目），演员（角色）们按照剧本（代码）的描述在舞台上进行各种表演（运动、对话、改变外观等）。

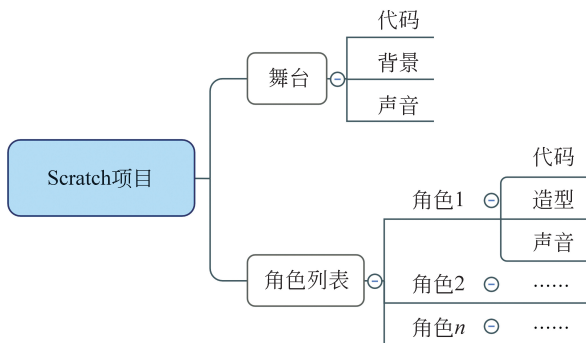


图 1-1-9 Scratch 项目的构成要素

Scratch 中的角色通常包括代码（Code）、造型（Costume）和声音（Sound）3 个组成部分。在创作 Scratch 项目时，主要工作就是给角色设计合适的造型，并编写控制角色行为的代码。

所谓代码（也称为脚本），就是使用 Scratch 的功能积木按照一定的逻辑关系组合而成的指令集合。每个角色都拥有自己的代码，它让角色获得“思想”，使角色能够以各自的方式在舞台中运动、变化或与其他角色和用户进行交互等。代码是 Scratch 项目中最重要的一部分，就像是一场正在演出的话剧（项目），如果某个演员（角色）弄错了自己的剧本（代码），那么就会导致演出失败。严格来说，Scratch 编程指的是使用 Scratch 提供的各种指令积木构建舞台或角色的控制代码。

所谓造型，其实就是一个图像，Scratch 支持 PNG、SVG、GIF、BMP 和 JPG 等多种格式的图片作为造型。例如，使用支持透明效果的 PNG 图片作为造型，可使角色自然融合于舞台背景之中；使用支持无损缩放的 SVG 矢量图片作为造型，可保证角色在舞台中被放大而不会失真。一个角色可以拥有多个造型，可以在项目运行中通过代码切换为不同的造型，但在同一时刻只能使用一个造型。角色的造型犹如演员的服饰，演员在表演话剧时按照剧情需要更换不同的服饰。

一个角色可以拥有自己的声音，在项目运行中通过代码播放声音效果，可使角色显得活灵活现。Scratch 支持使用 MP3、WAV、AU 和 AIF 等多种格式的音频文件为角色添加声音，也可以给舞台添加声音。

Scratch 的舞台是一个封闭的矩形区域，它提供了一个给角色活动的虚拟世界，被代码控制的角色能够在舞台中运动或者与其他角色和用户交互等。一个 Scratch 项目只有一个舞台，舞台可以拥有自己的代码，拥有多个声音和多个背景。与话剧在表演中可以根据剧本要求更换舞台的背景一样，可以在 Scratch 项目运行中通过代码为舞台切换不同的背景，但在同一时刻，在舞台上只能显示一个背景。背景其实就是一个图像，Scratch 支持多种格式的图片作为舞台的背景。

1.2 准备 Scratch 开发环境

1.2.1 安装 Scratch 离线编辑器

为了能够流畅地运行 Scratch 3 软件，建议使用的操作系统是 Windows 10 以上版本，或者 macOS 10.13 以上版本。Scratch 3 软件有在线版和离线版两种形式。离线版可以摆脱网络的制约，只要将 Scratch 3 软件安装到本地磁盘上，就可以随时创作自己的作品，这种方式更适合国内用户。下面介绍两种安装 Scratch 3 离线编辑器软件的方式。

1. 通过应用商店安装 Scratch 3 离线编辑器软件

如果使用的是微软 Windows 10 操作系统，可以到微软应用商店（Microsoft Store）中搜索 scratch，然后在搜索结果中找到 Scratch Desktop 软件，如图 1-2-1 所示，单击“获取”按钮即可将最新版本的 Scratch 3 软件安装到计算机中。安装成功之后，可以在桌面或者开始菜单中找到 Scratch 3 的启动图标。

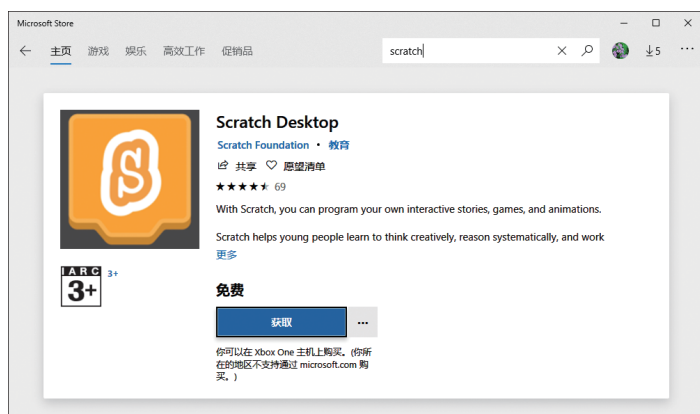


图 1-2-1 通过微软应用商店安装 Scratch 软件

如果使用的是 macOS 操作系统，可以到苹果应用商店（App Store）中搜索 scratch，然后在搜索结果中找到 Scratch Desktop 软件，如图 1-2-2 所示，单击“获取”按钮即可将最新版本的 Scratch 3 软件安装到计算机中。安装成功之后，可以在启动台中找到 Scratch 3 的启动图标。



图 1-2-2 通过苹果应用商店安装 Scratch 软件

2. 下载软件包安装 Scratch 3 离线编辑器软件

通过微软应用商店或者苹果应用商店可以安装最新版本的 Scratch 3 软件。如果需要安装某个特定版本的 Scratch 3 软件，可以访问微信公众号“小海豚科学馆”并发送消息 scratch，

就可以获取各个版本的 Scratch 软件包的下载链接。下载时请注意区分 Scratch 3 软件包文件的后缀名，后缀名为 .exe 的文件用于 Windows 操作系统，后缀名为 .dmg 的文件用于 macOS 操作系统。请根据自己使用的操作系统进行选择。

例如，在 Windows 7 操作系统中安装版本号为 3.25.0 的 Scratch 3 软件，可以先将安装包文件 Scratch Desktop 3.25.0 Setup.exe 下载到本地磁盘上，然后在资源管理器中双击文件名就可以启动 Scratch 3 软件的安装进程，接下来按照屏幕提示操作即可将 Scratch 3 软件安装到操作系统中。Scratch 3 软件的安装过程见图 1-2-3~ 图 1-2-6。

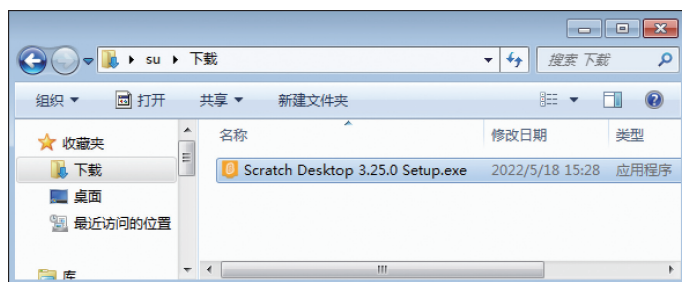


图 1-2-3 在资源管理中双击 Scratch 3 安装文件

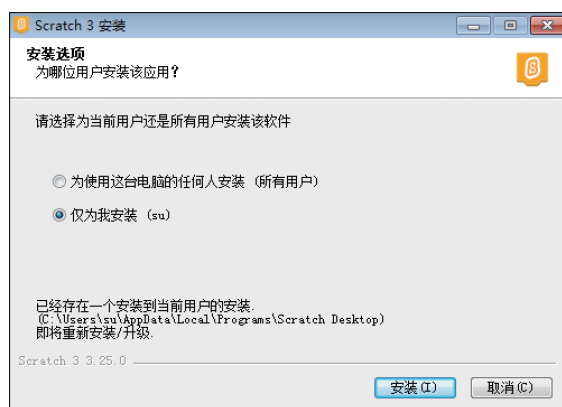


图 1-2-4 单击“安装”按钮开始安装进程

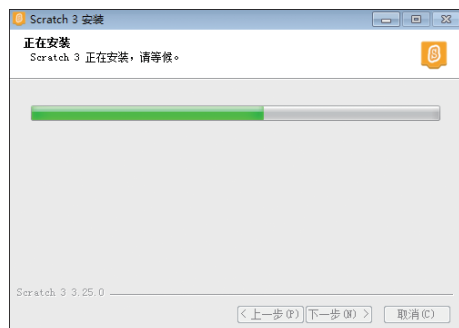


图 1-2-5 等待 Scratch 3 软件安装完成



图 1-2-6 单击“完成”按钮结束安装进程

Scratch 3 软件安装完成后,可以在 Windows 桌面和开始菜单中找到该软件的启动图标。Scratch 3 软件运行之后,其初始界面如图 1-2-7 所示。引人注目的是在屏幕右侧的舞台上有一只可爱的小猫,在今后的编程学习中,我们将经常和这只小猫打交道。

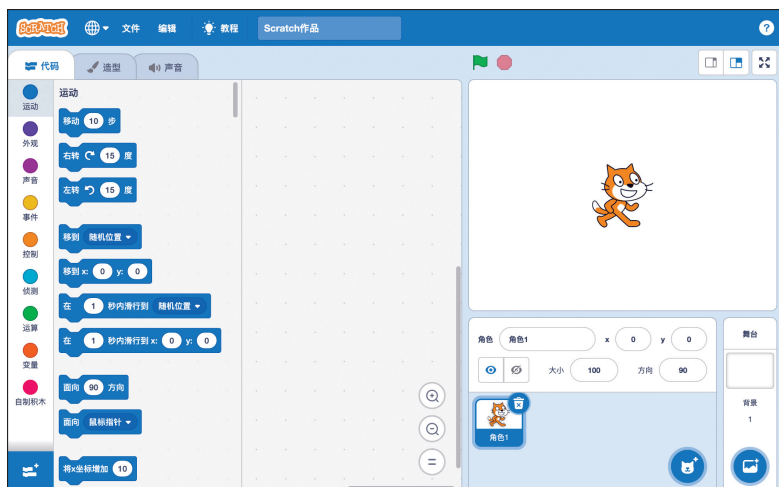


图 1-2-7 Scratch 3 软件运行界面

1.2.2 访问 Scratch 在线编辑器

在能够顺畅访问互联网的情况下,通过网络浏览器(如 Firefox、Chrome 或 Safari 等)就可以便捷地使用 Scratch 3 在线编辑器创作项目,不需要在用户的计算机中下载和安装 Scratch 3 软件。

在 CodeLab Scratch 编程社区网站中提供 Scratch 3 在线编辑器的使用服务,访问网址 <https://create.codelab.club> 即可进入社区主页。如图 1-2-8 所示,在社区主页面单击“开始创作”按钮就可以启动 Scratch 在线编辑器,之后就可以像使用离线编辑器一样创作各种有趣的作品。



图 1-2-8 CodeLab Scratch 社区主页