

CodeLab 手册

CodeLab 手册记录了 CodeLab 在探索计算机编程过程中的想法、实践和工具，希望能提供一些不同的视角。

了解 CodeLab

- [玩给你看](#)
- [CodeLab 主页](#)
- [关于 CodeLab](#)

编程学习

课程

为了帮助编程初学者入门 Scratch 和 Python，CodeLab 设计或整理了 4 门课程：

- Scratch
 - [【创意编程】](#)
 - [【创意计算】](#)
 - [【Scratch 编程工作坊】](#)
- Python
 - [【交互计算】](#)

学习路径

在开始学编程前，先问问自己为什么要学编程？如果你想探索编程的乐趣，更好地表达自己的创意或想法，建议从 Scratch 开始。可以先体验 [Scratch 入门项目](#)和 [CodeLab 社区精选作品](#)，对 Scratch 能做什么有了初步了解后，可以学习 [【创意编程】](#)或 [【创意计算】](#)课程。[【创意编程】](#)包含视频和学习讲义，适合自学。[【创意计算】](#)则适合在有老师指导的情况下和同伴一起学习。这两门课都是 Scratch 入门课程，任选其一学习即可。如果想开展 Scratch 编程入门的工作坊，可以使用 Scratch 官方设计的[活动指南](#)。[【Scratch 编程工作坊】](#)则是针对有一定 Scratch 编程基础的学习者，比如学过[【创意编程】](#)或[【创意计算】](#)课程。Scratch 入门后，可以经常逛逛 [CodeLab 社区](#)，看看大家创作的作品，从社区中找到感兴趣的项目学习，通过改编作品、组建[工作室](#)和同伴一起创作，表达自己的想法，探索编程的乐趣。

如果想继续拓展 Scratch 创作的边界，可以学习 [【交互计算】](#)课程，它是一门使用 Scratch 和 Python 共同创作作品的 Python 入门课程。Python 非常擅长处理数据，并且有海量的第三方库来扩展 Python 的能力。学习编程有各种不同的路径，如果你对编程能做什么还不太了解，可以参考以上的学习路径。如果你有明确的兴趣或需求，那么依照你的兴趣或需求去学习效果可能更好。

CodeLab 推荐

CodeLab 推荐的工具与资源。

编程环境

- Scratch
 - [CodeLab 创作平台](#)
 - [Scratch 高级编辑器](#)
- Python
 - [mu-editor](#)
 - [JupyterLab](#)

课程资源

- [UCL ScratchMaths Curriculum](#)(由社区用户 @kwyjibo 分享)
- [CS50](#)
- [Elements of AI](#)
- Python 书籍推荐
 - [Head First Programming](#)(深入浅出程序设计)
 - [A Byte of Python](#)(Python 简明教程)
- Python 参考手册:
 - [python-cheatsheet](#)
 - [Python 文档](#)
 - [Python 学习手册](#)
- [griffpatch Scratch Tutorials](#)

创作工具

- [Chrome Music Lab](#)

计算机历史与文化

- [A Computer for Children of All Ages](#)
- [Personal Dynamic Media](#)
- [User Interface: A Personal View](#)
- [《黑客: 计算机革命的英雄》](#)
- [Learnable Programming](#)
- [A Brief Rant on the Future of Interaction Design](#)
- [What Can A Technologist Do About Climate Change?](#)
- [Dynamicland-501c3-Narrative](#)

附录

- [CodeLab 社区](#)
- [CodeLab 文档](#)
- [CodeLab 博客](#)