

PROGRAMMING WITH SCRATCH

Most people view **computer programming** as a specialized activity for the technological elite, appropriate for only a small segment of the population. And, indeed, traditional programming languages like Java and C++ are difficult for most people to learn.

Scratch takes advantage of the extraordinary advances in computing power, networking, and interface design of the past two decades to make programming more accessible and engaging for children, teens, and others who are learning to program. Key features of Scratch include:

- **Building-block programming.** To create programs in Scratch, you simply snap graphical blocks together into stacks. The blocks are designed to fit together only in ways that make syntactic sense, so there are no syntax errors. Different data types have different shapes, eliminating type mismatches. You can make changes to stacks even as programs are running, so it is easy to experiment with new ideas incrementally and iteratively.



- **Media manipulation.** With Scratch, you can create programs that control and mix graphics, animations, music, and sound. Scratch extends the media-manipulation activities that are popular in today's culture – for example, adding programmability to Photoshop-style image filtering.



- **Sharing and collaboration.** The Scratch website provides inspiration and audience: you can try out other people's projects, re-use and adapt their images and scripts, and post your own projects. The ultimate goal is to develop a shared community and culture around Scratch.



Scratch offers a **low floor** (easy to get started), **high ceiling** (ability to create complex projects), and **wide walls** (support for a wide diversity of projects). In developing Scratch, we put high priority on simplicity, sometimes even sacrificing functionality for understandability.

As students work on Scratch projects, they have opportunities to learn important **computational concepts** such as iteration, conditionals, variables, data types, events, and processes. Scratch has been used to introduce these concepts to students of many different ages, from elementary school through college. Some students transition to traditional text-based languages after getting introduced to programming with Scratch.

Scratch is built on top of the **Squeak** programming language. It was inspired by previous work on **Logo** and Squeak **Etoys**, but it aims to be simpler and more intuitive.

Scratch is an **open-source** but **closed-development** project. The source code is freely available, but the application is developed by a small team of researchers at the MIT Media Lab.