UNIT 4
OVERVIEW

THE “BIG IDEA”

Personalization is an important guiding principle in the design of the creative computing experience. By “personalization”, we mean both connecting to personal interests and acknowledging that personal interests can vary considerably. There are many ways of knowing and doing – and exploring these multiple ways can help support interest, motivation, and persistence among young learners. In this unit, learners explore some of the advanced concepts and challenging problems associated with game design. An advanced concept or challenging problem can be made more accessible if rooted in activities that are personally meaningful. As an example of the power of context, we turn to a story shared by Mitch Resnick – the director of the Scratch project at MIT.

A few years ago I was at one of our Computer Clubhouse after school centers and I saw a 13-year-old boy working on creating his own game. He was able to control a character, in this case, a fish. He wanted the game to keep track of the score, so you could see how many little fish had been eaten by the big fish, but he didn’t know how.

I saw this as an opportunity to introduce the idea of variables. I showed this to him and he immediately saw how he could use this block to keep track of how many fish had been eaten in his game. He took the block and put it in the script right where the big fish eats the little fish. He quickly tried it. Sure enough, every time the big fish ate a little fish, the score goes up by 1.

I think that he really got a deep understanding of variables because he really wanted to make use of it. That’s one of our overall goals of Scratch. It’s not just about variables, but for all types of concepts. We see that kids get a much deeper understanding of the concepts they learn when they are making use of the concepts in a meaningful and motivating way.

LEARNING OBJECTIVES

Students will:
+ be introduced to the computational concepts of conditionals, operators, and data (variables and lists)
+ become more familiar with the computational practices of experimenting and iterating, testing and debugging, reusing and remixing, and abstracting and modularizing by building and extending a self-directed maze, pong, or scrolling game project
+ identify and understand common game mechanics

KEY WORDS, CONCEPTS, & PRACTICES

+ abstracting and modularizing
+ conditionals
+ operators
+ data
+ variables and lists
+ sensing
+ feedback fair
+ arcade day
+ puzzle jar
+ brain dump

NOTES

+ Many new concepts are explored in this unit, so we’ve included added support in the form of example project studios, new programming puzzles for extra practice, and starter game projects that we encourage you to remix and reuse as needed.
In this unit, learners will become game designers and experience creating their own game project. Guided by the activities in this unit, students will be introduced to game mechanics and game development while building understandings of computational concepts (conditionals, operators, data) and computational practices (abstracting and modularizing).

You could get students started on their game projects with the Starter Games activity and then support further development through other activities. From learning common game mechanics such as keeping score and side-scrolling, to the creation of multiplayer games (e.g., Pong), Unit 4 activities offer students multiple opportunities to practice game development.