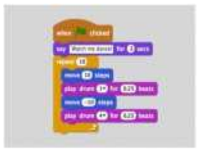
Milton Public Schools

Grade Three STEM – Creative Computing

What is Creative Computing?

Creative computing is about creativity. Computer science and computing-related fields have long been introduced to young people in a way that is disconnected from their interests and values - emphasizing technical detail over creative potential. Creative computing supports the development of personal connections to computing, by drawing upon creativity, imagination, and interests.

Creative computing is about empowerment. Many young people with access to computers participate as consumers, rather than designers or creators. Creative computing emphasizes the knowledge, practices, and fundamental literacies that young people need to create the types of dynamic and interactive computational media that they enjoy in their daily lives.



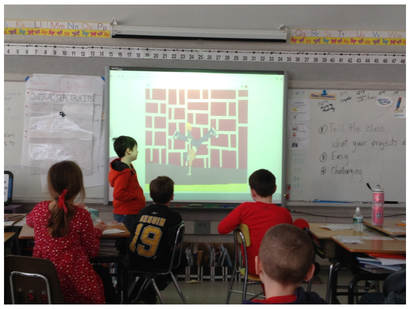
Creative computing is about computing. Engaging in the creation of computational artifacts prepares young people for more than careers as computer scientists or programmers. It supports young people's development as computational thinkers - individuals who can draw on computational concepts, practices, and perspectives in all aspects of their lives, across disciplines and contexts.

What is Scratch?

There are many different tools that can be used for creative computing. In third grade in the Milton Public Schools, we use Scratch, which is a free computer programming language developed by researchers at the MIT Media Lab. With Scratch, people can create a wide variety of interactive media projects - animations, stories, games, and more - and share those projects with others in an online community. Since Scratch's launch in May 2007, hundreds of thousands of people all around the world have created and shared more than 6 million projects.

What will students learn?

Students will engage in activities during an introductory creative computing experience using the Scratch programming language. The activities are designed to support familiarity and increasing fluency with computational creativity and computational thinking. In particular, the activities encourage exploration of key computational thinking concepts (sequence, loops, parallelism, events, conditionals, operators, data) and key computational thinking practices (experimenting and iterating, testing and debugging, reusing and remixing, abstracting and modularizing).



The Creative Curriculum Guide (written by Karen Brennan, Michelle Chung, and Christan Balch of Harvard) provided this information. Visit http://scratched.gse.harvard.edu/guide/ for more information.