SqueakCMI Notebook:
Projects, Tools, and Techniques

Introduction

Welcome to eToys/Squeak: an object-oriented programming language. This notebook was written to introduce Squeak to curious beginners with step-by-step descriptions of projects and how they were done.

Advice is freely given in the hope that the path you take to learning eToys/Squeak is quick and smooth. The Squeak community will be generous with their time, their knowledge, and their willingness to help newcomers. The Office for Mathematics, Science, and Technology Education at the University of Illinois Urbana-Champaign invites you to use these materials to the benefit of students everywhere.

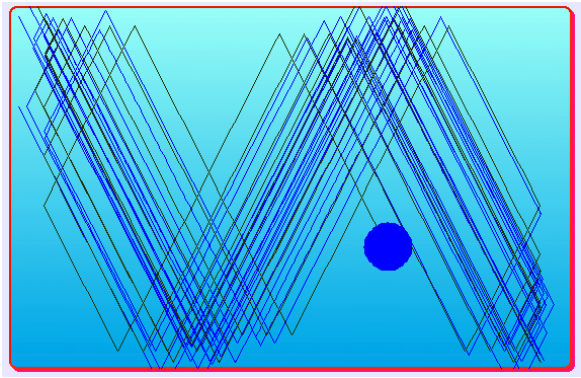
These projects can be explored on the computer by opening them from www.Squeakcmi.org. This dynamic experience of projects on the computer in conjunction with the written materials should give you a range of ideas and possibilities to combine in many ways and for many purposes.

Section I

This section contains two easy projects designed to help you get started with Squeak. They are followed by an extensive description of the rich resources, tools, icons, supplies, and conventions that make Squeak what it is.

Section II

This section explains more complicated projects. They are in alphabetical order by the name of a Squeak tool used predominantly in that project. The projects are not in sequential order by level of difficulty. The project's name can be used to locate that project atwww.Squeakcmi.org. So, if you wonder, "What is a scale factor and how could it be used in a Squeak project?" you can find out.



www.SqueakCMI.org
Resources, projects, tutorials, and standards-based lessons applying Squeak in math, science, language arts, social science, and art. Additional projects and essays can be found on the website. Tutorials developed by math specialists show the myriad ways Squeak enriches the study of geometry and trigonometry. The SqueakCMI community can answer questions, share ideas, and schedule workshops.

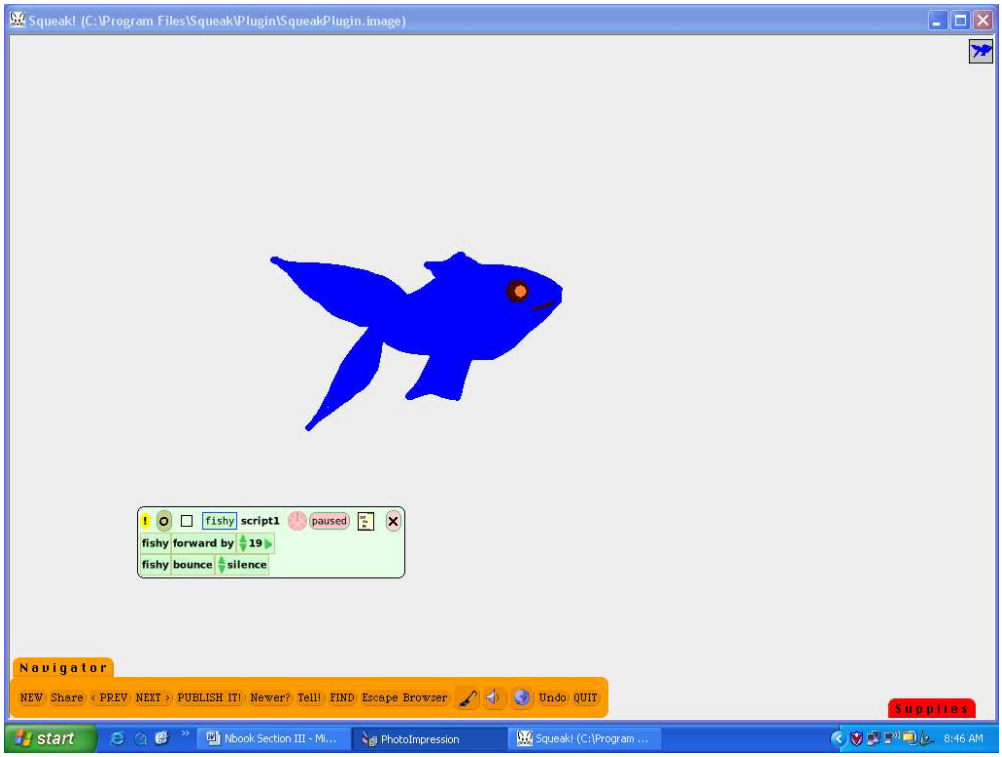
www.squeakland.org
The origin of Squeak: software, tutorials, and example projects. Get the most current versions of the software at Squeakland. The site includes interesting essays about the nature of learning, about programming and thinking.

www.squeak.org
Technical information for experienced programmers and developers

Kathleen Harness
squeakcmi@uiuc.edu

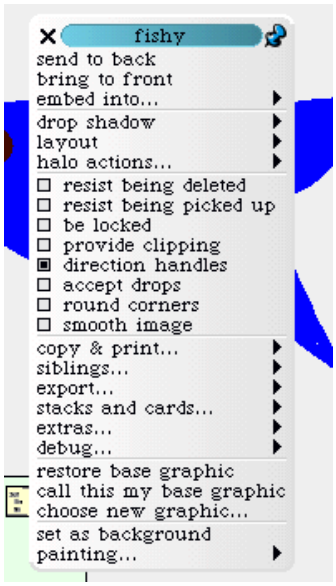
Set Rotation Style: fishySwimnb

This project shows how to change the rotation style of an object when it bounces against the edge of the world screen or in a playfield. The fish in the project will bounce and turn upside down unless the rotation style is changed so that it flips left to right.

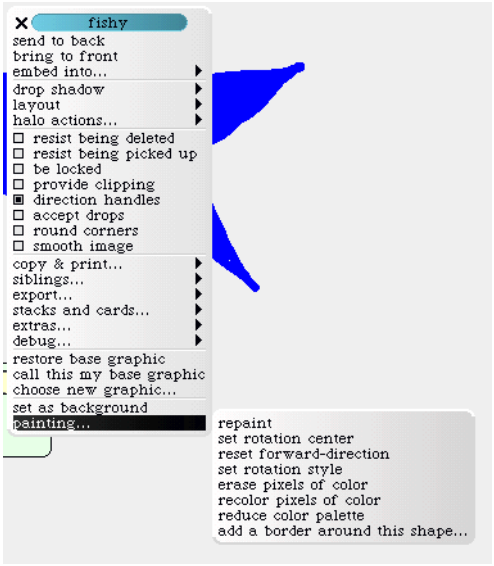


Draw an object and keep it and then get the halo of handles; choose the red halo and open a menu of choices.

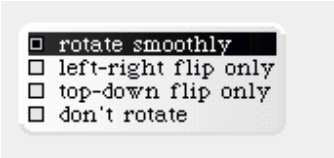
The halo's red handle menu looks like this:



When this menu is open, click on the little push pin symbol and the menu will stay open while you read it. At the very bottom of the list is painting and a little right pointing arrow indicates there are more choices for that item in the list. Click on painting to open the next list and choose set rotation style to open one last list in this long path to a useful place.

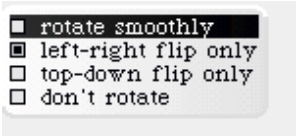


Set rotation style shows four choices and the top one is marked as the default choice. A default is just a basic setting which can be changed though these kinds of menus so that your project's objects, fish or birds or cars move in the way you want them to move. You can choose to make it more realistic or more imaginary, you decide.



The fish in this project will move forward by five back and forth across the screen with a heading of 90. The script includes bounce from the motion category of script panels. With the default choice (rotate smoothly) the fish will be upside down as it swims in one direction.

Using the set rotation style menu and choosing left-right flip only will make the fish swim upright in both directions.



While you have this menu open try the other choices so that you develop a vocabulary of how these commands will work. You might want them in some other scripts.