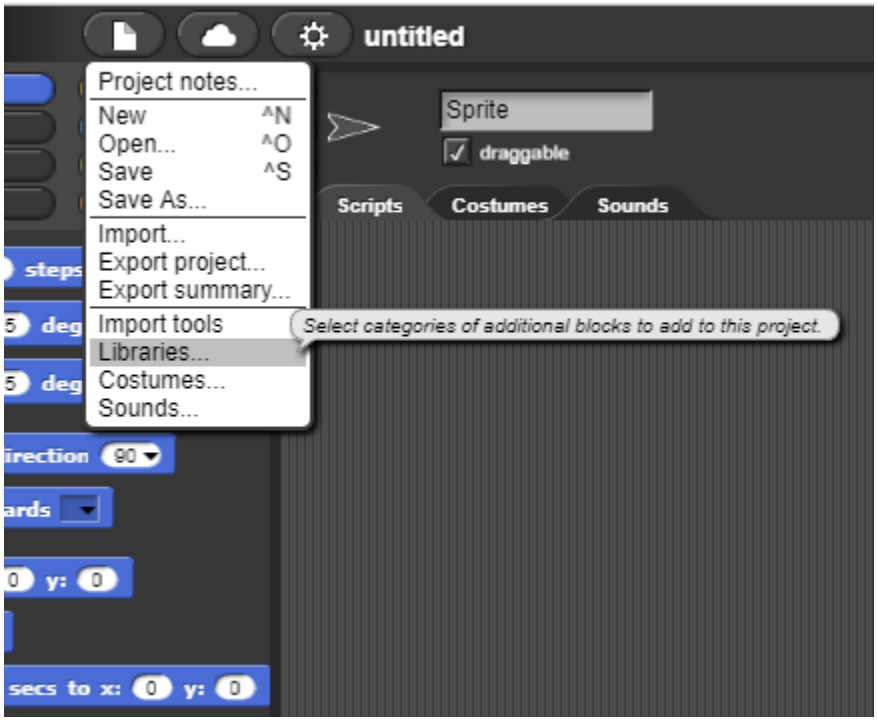
Discussion 1: Snap! Scavenger Hunt Solutions

INSTRUCTIONS: The questions below are designed to highlight some important Snap! features and clarify some common misconceptions. You should open up Snap! in your browser and experiment as you try to determine the answers. You should also "Import Tools" before you begin (ask your TA if you need help with this).

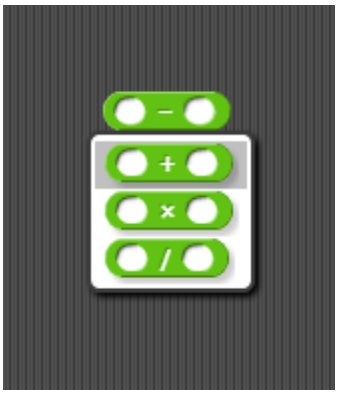
1. What are "libraries," how do you import them, and why might they be useful?

Libraries are collections of pre-made blocks that are <u>not</u> automatically loaded when you open Snap!. Typically, a library's blocks are all oriented toward accomplishing a particular task or assisting with a particular aspect of programming. Some commonly used Snap! libraries are "List Utilities," "Words & Sentences," and "Iteration & Composition."

Libraries offer you the benefits of custom-made blocks without the strain of having to implement those blocks yourself. You can just take advantage of the work someone else has already put in. And, in keeping with the principle of abstraction, you <u>don't</u> need to understand how the blocks are implemented in order to use them effectively.

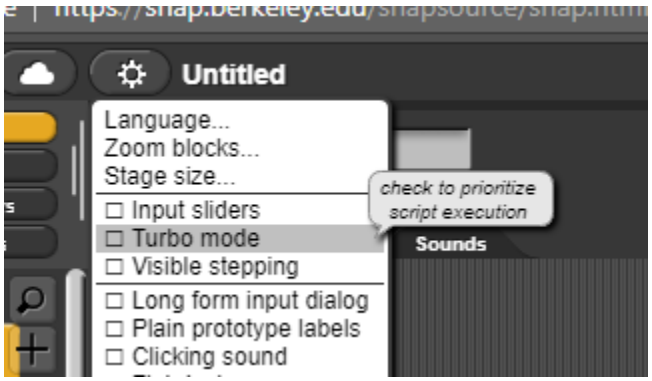


2. You are writing code to calculate your grade in CS10, when you realize that you've been using the wrong block. Instead of subtracting two numbers, you should be adding them together. What do you do to *relabel* the "subtract" block as an "add" block?



3. How do you find out what the function below does without writing any code?

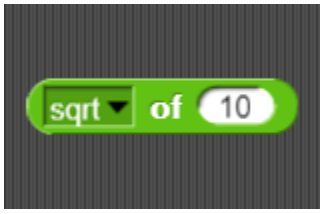
4. You are trying to run a program in Snap!, but it's taking *forever* to execute. How can you make this program run faster? (Hint: There are two features in Snap! that would allow you to do this.)

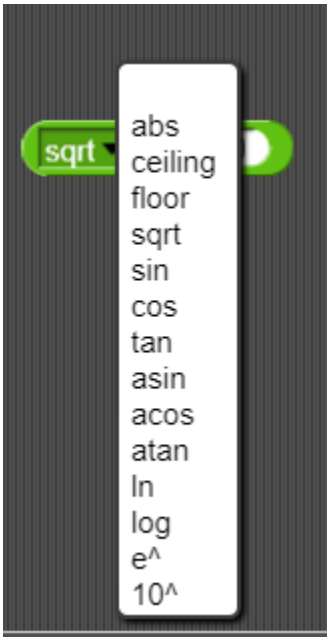


5. Which symbol out of {!, @, #, $, %, ^, &, *, ~} helps you create a variable as you are typing the name of a custom block?

%, so for example, %x + %y would create the two variables x and y automatically.

6. To help your friend with their math homework, you want to find the tangent of 0. Which block in Snap! allows you to do that? (Hint: Look for a block that lets you do a lot of different mathematical operations….)



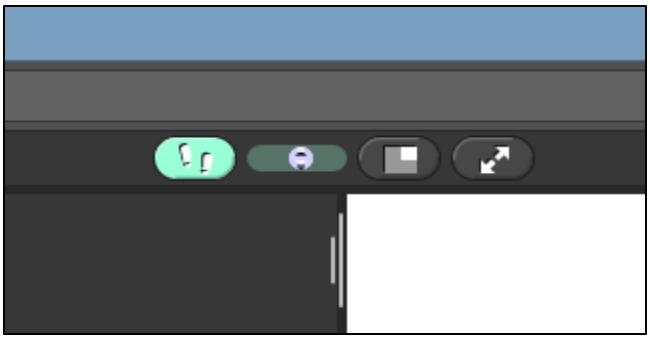


7. You're happily drawing images in Snap! when your sprite drifts off the stage and disappears! Uh-oh! How do you get your sprite back onto the stage?

8. You've been trying to debug your code for the past three hours, but still can't figure out what's wrong. What tool in Snap! can you use to help you find the bug?

Turn on "visible stepping." This is an incredibly useful feature of Snap! that allows you to step through your code block-by-block. As each block executes, it becomes temporarily highlighted. You can also adjust the speed to make the stepping easier to follow.

The screenshot below indicates how to turn on visible stepping. We highly encourage you to test out this feature now. It is bound to be very helpful at some point in the semester.



9. For each of the tasks listed below, find a Snap! block that accomplishes it.

- Return the *opposite* of True or False.

- Check if the variable “mystery” is set to text or a number.

- Ask a question in Snap and automatically save the answer into a variable called “answer.”

- Remove a Sprite from the stage (*without* deleting the Sprite entirely).

- Make a sprite *write,* not say, “BJC” with font size 20.

- Find out if the left mouse button is currently clicked.