Tiny Bubble Sort

Tiny Bubble Sort uses a *bubble sorting algorithm* to sort five pieces of data input by the user.

Create a background and sprite for decorative purposes only. In Data, create a variable, n, and a list, My List. The list will consist of exactly 5 items.

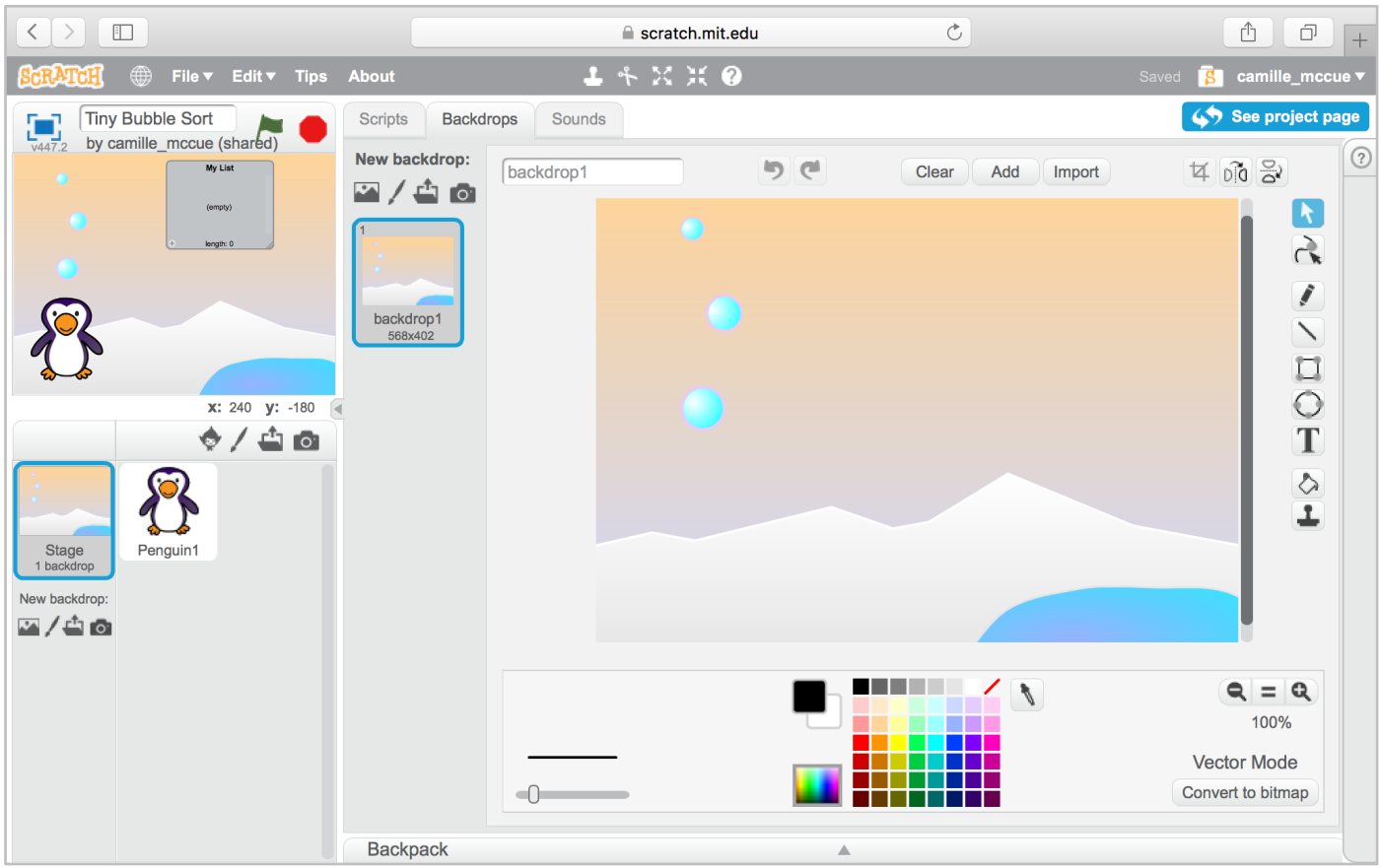
Write a green flag script to delete previous items from the list. Then ask the user to input, one at a time, 5 numbers. Each answer is added to the list.

Write a sort script, executed by pressing the "s" key, that compares the first and second items, and orders them from small to large. Repeat this process, moving on to the second and third items; then the third and fourth items, etc. Finally, repeat this pairwise sorting a total of four times to completely order the list from smallest to largest.

To start the action: Press the green flag.
To sort the list: Press the "s" key.

Tiny Bubble Sort

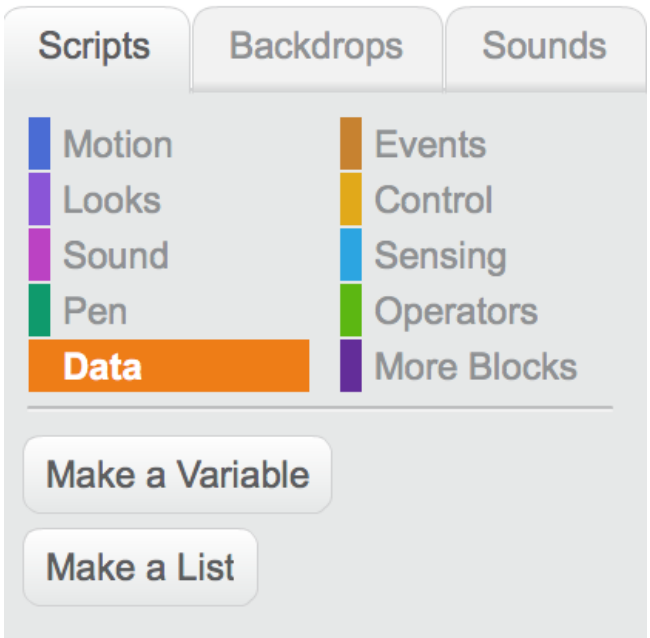
Stage – Backdrop

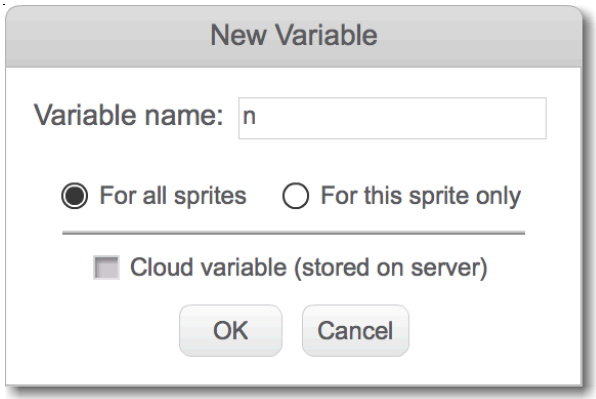


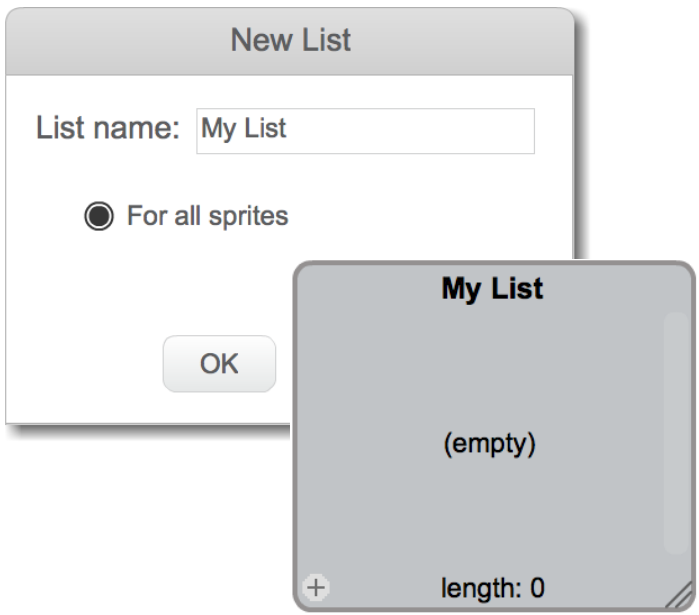
Penguin Sprite – Costume

Tiny Bubble Sort

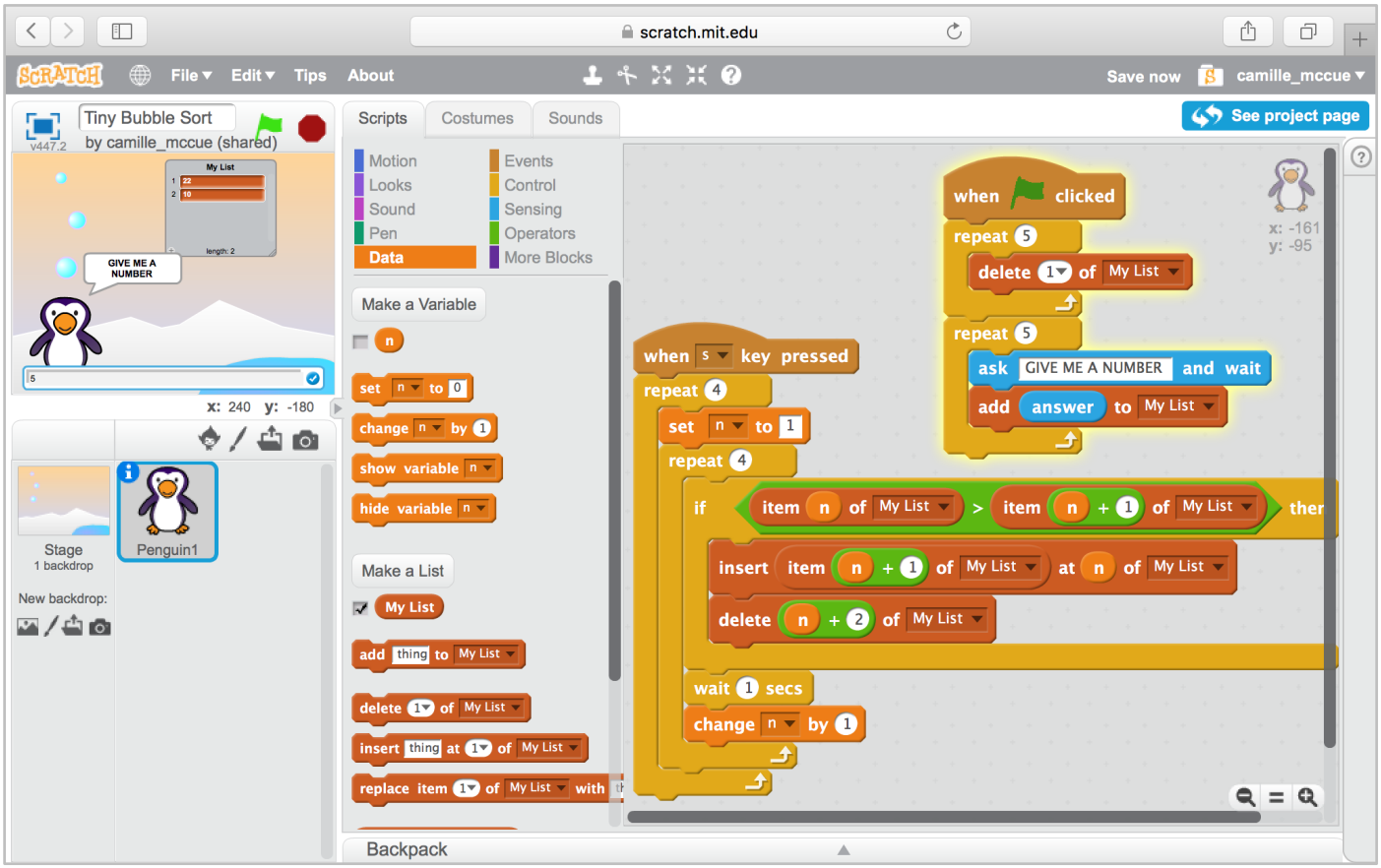
Penguin Sprite – Variable and List





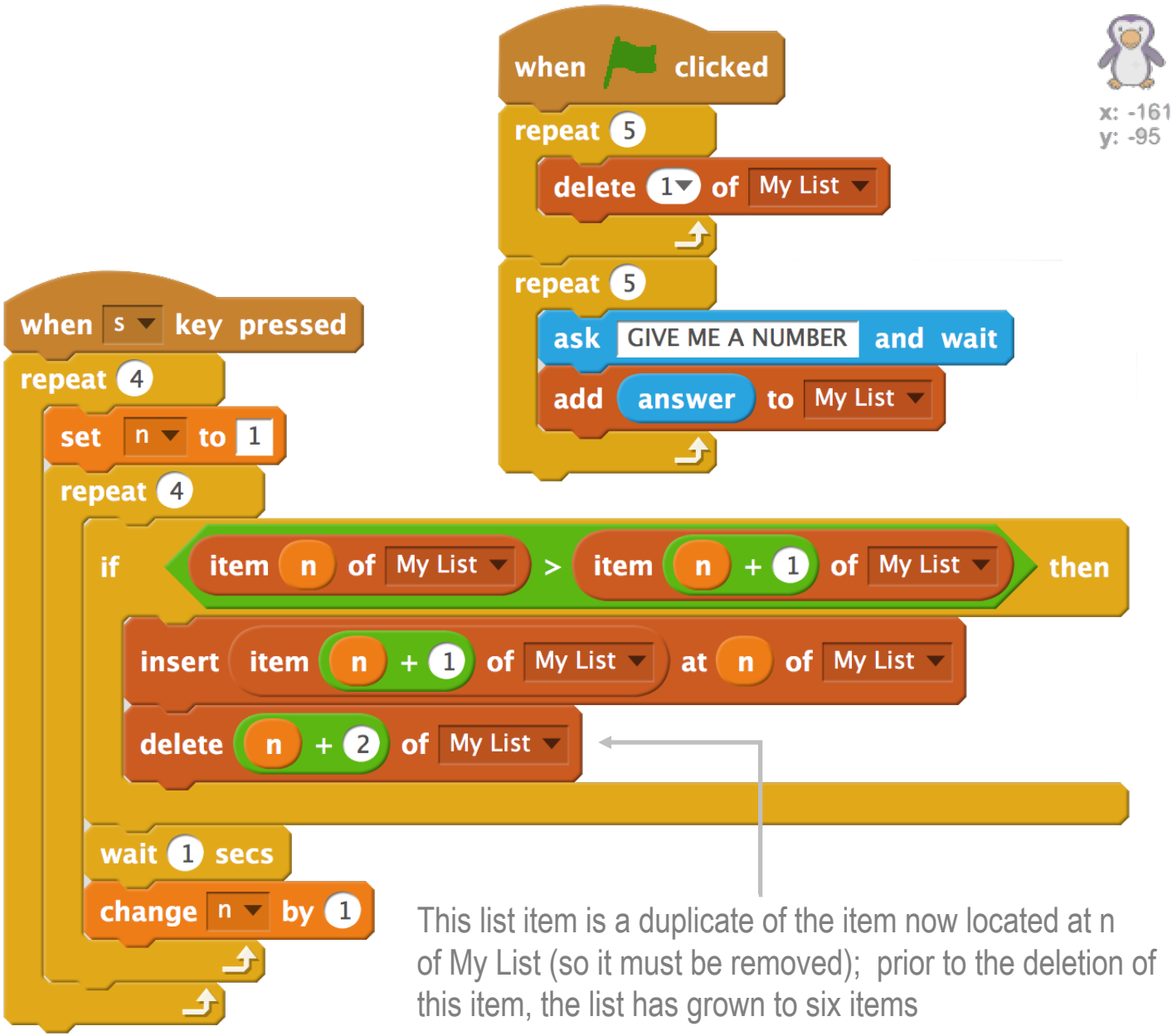


Penguin Sprite – Scripts



Tiny Bubble Sort

Penguin Sprite – Script Closeups



This list item is a duplicate of the item now located at n of My List (so it must be removed); prior to the deletion of this item, the list has grown to six items

Extension: Can you think of a way to, through abstraction, make this bubble sorting algorithm more general so that a user can vary the length of My List on every use of the app?

Algorithmic efficiency is the property of an algorithm which measures the amount of computational resources used by the algorithm. Better efficiency means faster processing time – something that may matter greatly when the data set is large. What is the algorithmic efficiency of a bubble sort? How does this compare with other sort methods – is it a "fast" or "slow" method?