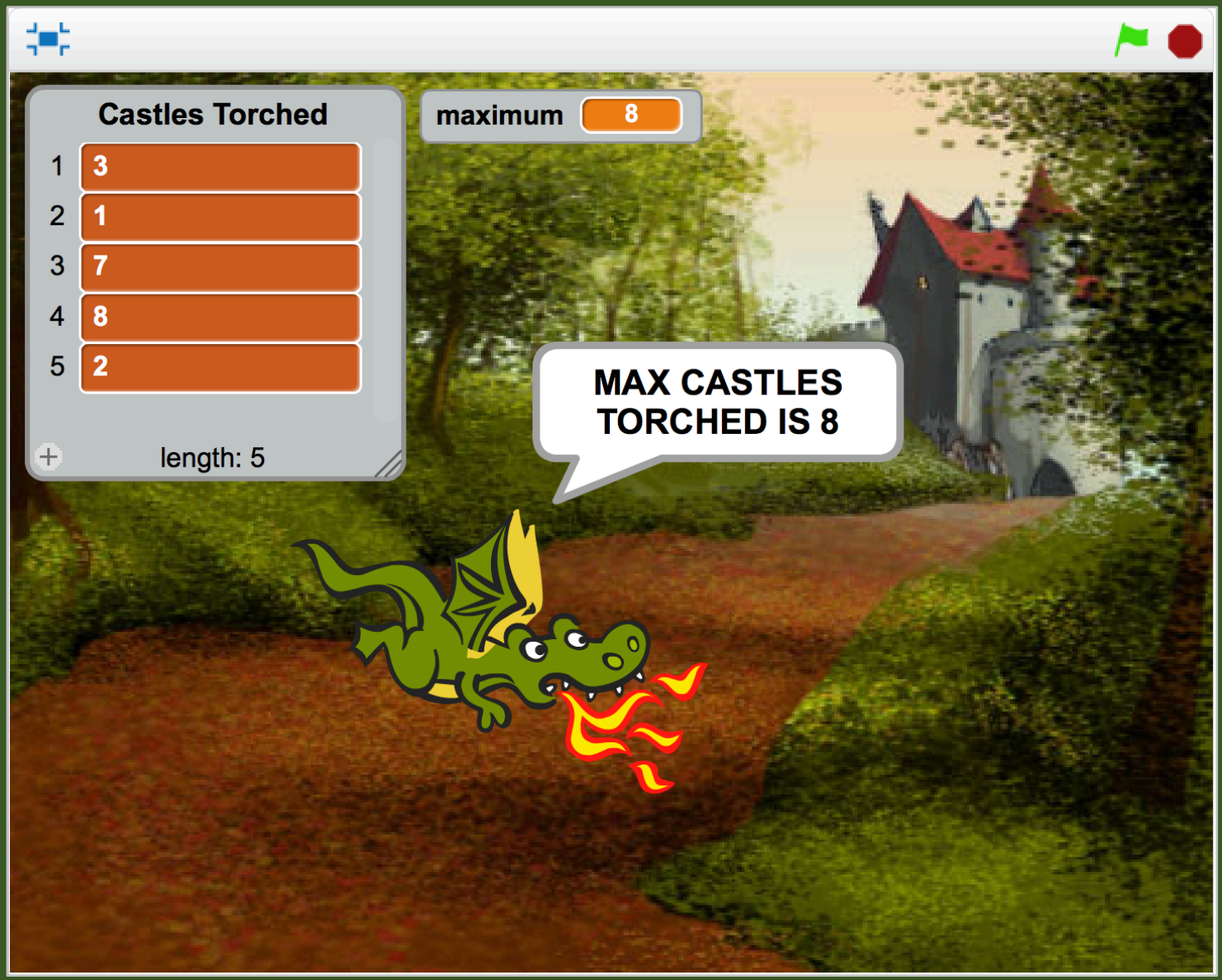
Torching Castles



Code It

Torching Castles is a text input/output app in which a dragon tells the user about castles torched by her fiery friends.

Add a castle backdrop to the stage and a dragon sprite. Then add a list (similar to a one-dimensional *array)* called Castles Torched – the user will be asked to tell the dragon how many castles each of her five friends torched. Also create a variable named maximum. Write a script that determines which list item is the maximum.

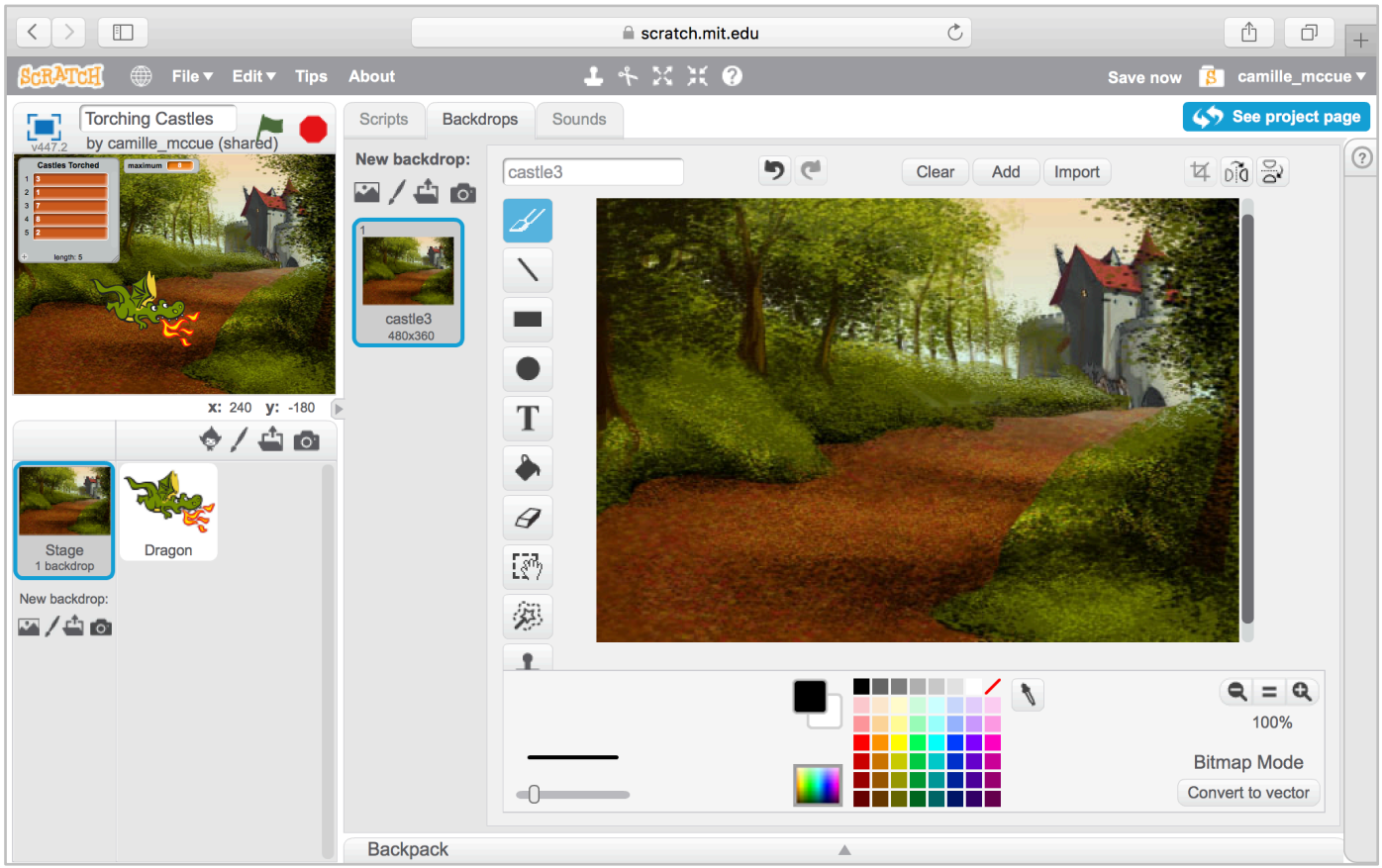
Start the action by initializing the list and variable (clearing list items and setting the maximum value to zero) with the green flag.

Play It

Press the green flag to start the action.

Torching Castles

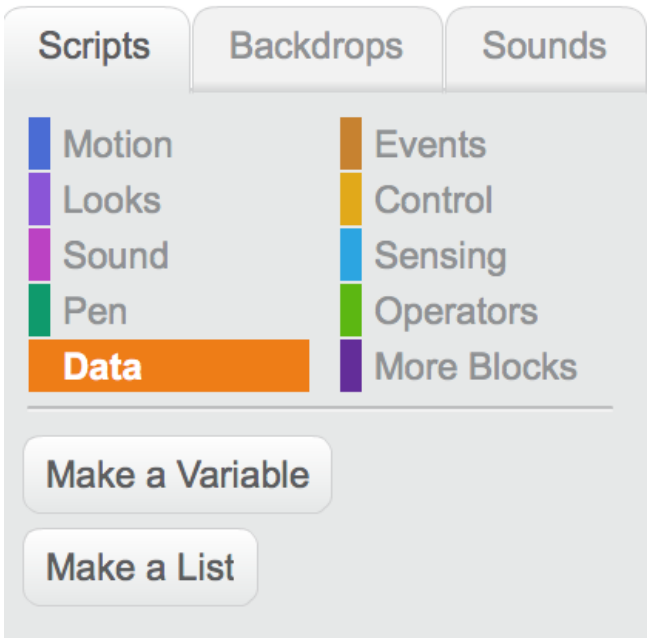
Stage – Backdrop

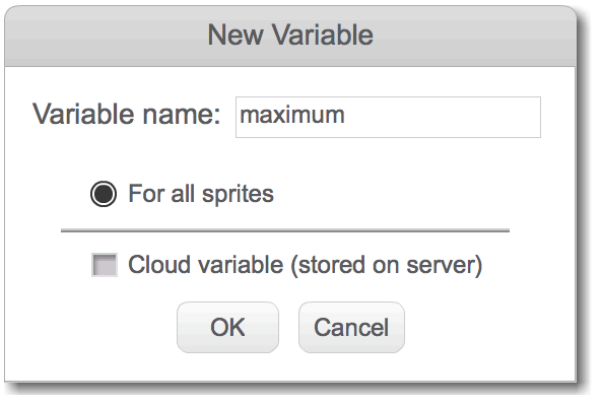


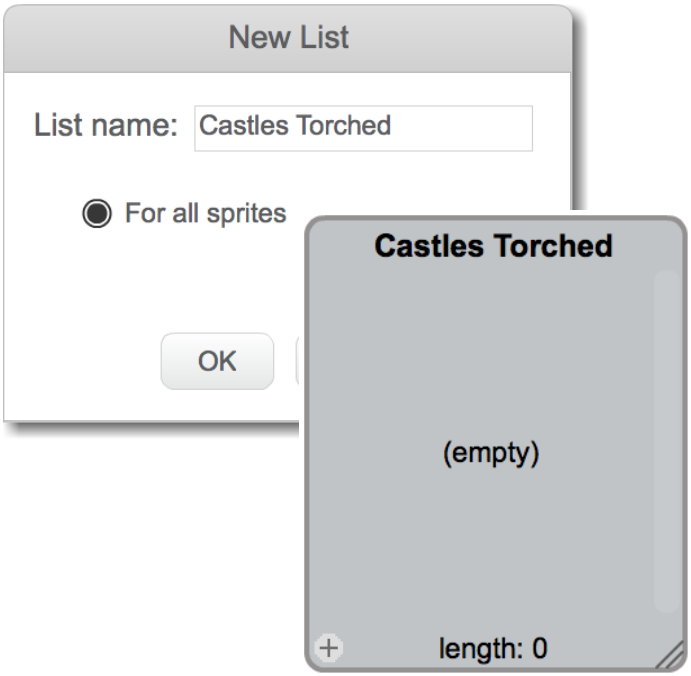
Dragon Sprite – Costumes

Torching Castles

Dragon Sprite – Variables



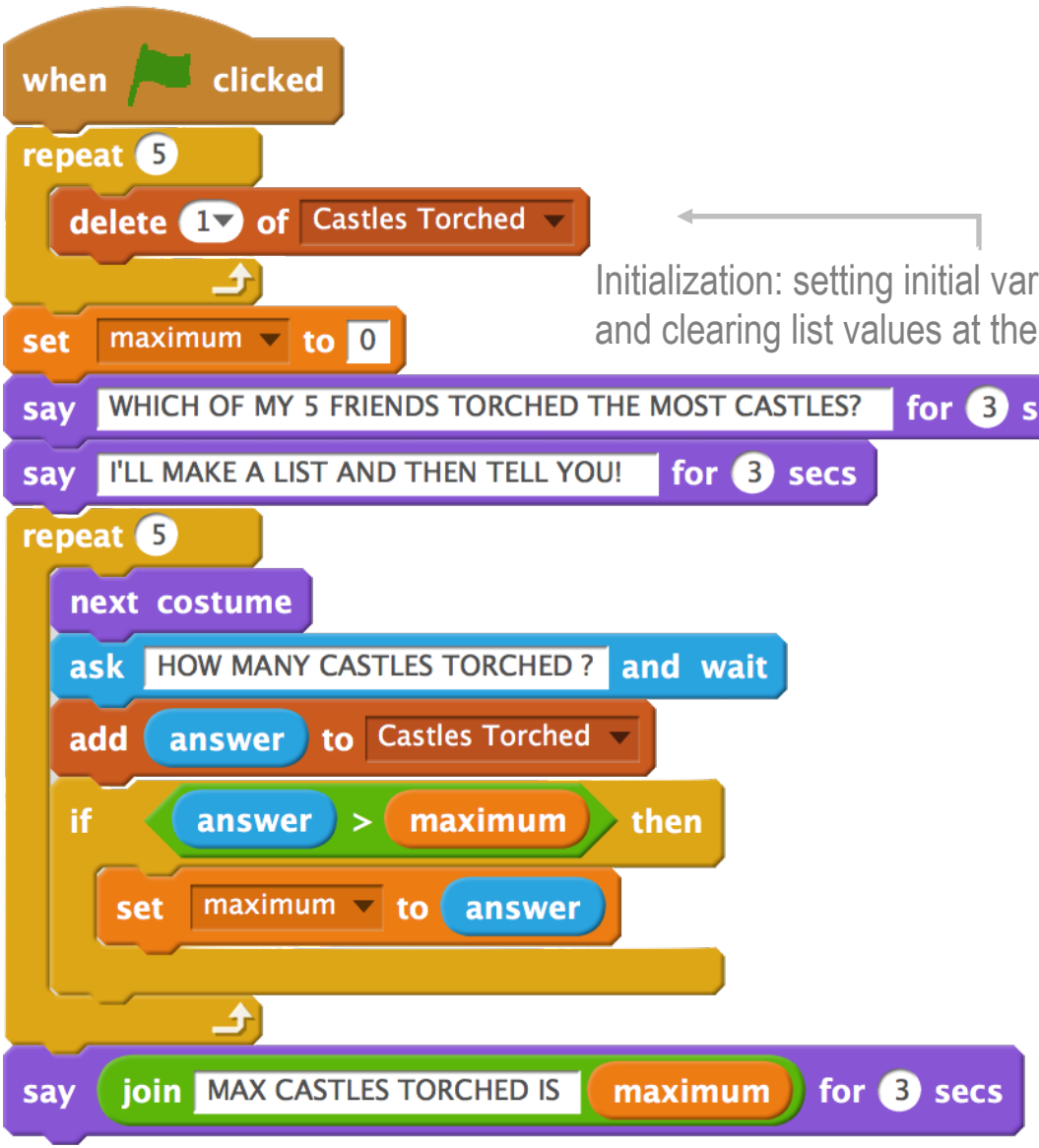




Dragon Sprite – Script

Torching Castles

Dragon Sprite – Script Closeup



Initialization: setting initial variables (often to zero) and clearing list values at the outset of a program

Extend It

Torching Castles features a specific number of castles, namely five. You wrote the program to handle this specific case.

Can you generalize this program to handle *n* number of castles, a number that may change on each execution of the program? For example, *ask* the user to enter numbers until a trigger letter, such as "x" is entered, indicating the user is done.

In computer programming, moving from a specific case (a list of five items) to a general case that is useful in more circumstances (lists of different lengths) can be considered one definition of abstraction. (Another definition of abstraction: you want to include all of the relevant information in the program, but none of the extraneous information.)