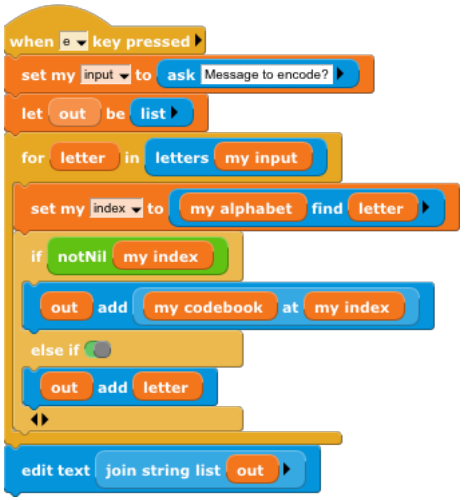
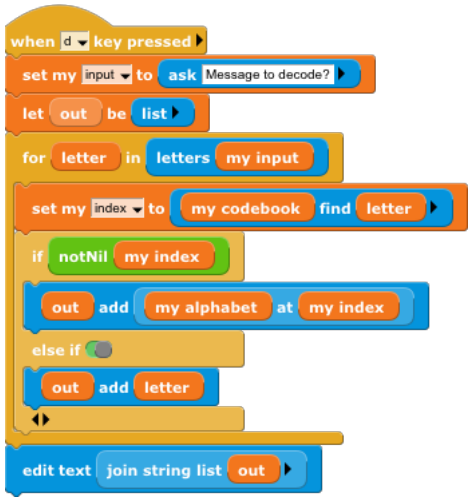




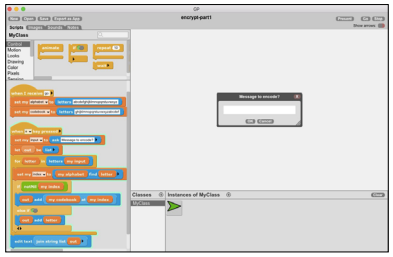
Preview of Part 2

In this project, we created two separate codes for encoding and decoding that are very similar. In Part 2, we use the principle of ***abstraction*** to make a single script that does both the encoding and the decoding. Here is a preview of some of the blocks we'll use in part 2!

GP Guide #1 | Encryption Part 1

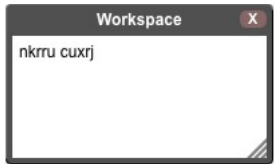
Overview

In this guide, you will make a project that will encode and decode secret messages.



How it works

This project uses two lists of letters for encoding. One list is the alphabet. The other list -- the codebook -- is some permutation of those letters; in this guide, it is the alphabet all mixed up! Here's how it works: when the letter ***h*** is typed in a secret message, the alphabet list locates ***h*** and then finds the corresponding character in the codebook. Since h is the eighth letter in the alphabet, the corresponding character in the codebook will be the eighth one in from the beginning of the list. This process is called indexing. Let's get started!



1 To get set up, create a variable for each index and add the letters for each list. The two lists must be exactly the same length and the codebook should not have any duplicated letters. This will ensure that the encoding and decoding of messages will process without error.

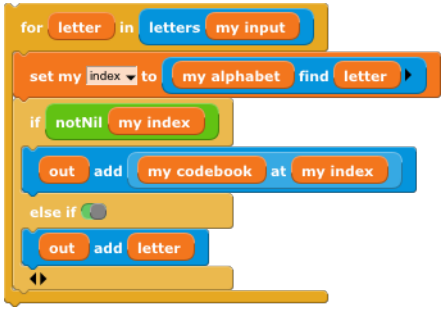
NOTE: there are only lowercase letters included in the alphabet list, however the codebook might include uppercase letters, numbers, symbols, or even emojis!

2 When the ***e*** key is pressed, the project will prompt the user to enter a message to *encode*. To get started, use the ***when key pressed*** block and ***set my input*** to ask for the user's message.

Use the ***let var be*** block to set that the out be list. Click on the orange ***var*** block to rename.

3 The next part of the script for the encoder uses a ***for loop***. The encoding process works by finding the index of the each letter in the alphabet list, then using that index to get the corresponding codebook letter.

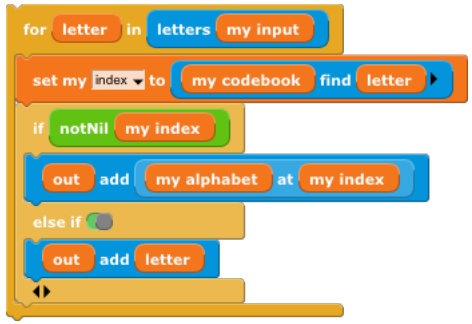
Each encoded letter is added to a list called ***out*** and at the end, all of those letters are joined together to make the output.



4 Next up, use the block called ***edit text*** to display the output. This allows the user to select the resulting text so they can copy and paste the encoded message and send to a friend or put it into the decoder.

5 The next part of this project is the decoder. When the ***d*** key is pressed, the project will prompt the user to enter a message to *decode*.

6 This ***for loop*** is similar to the one used for the encoder script. However, there is one key difference. The ***my codebook*** and ***my alphabet*** blocks are switched. This script will allow the user's message to be translated back into its original form.



7 Finally, use the block called ***edit text*** to display the output again. This will allow the user to see the original message!