

Lesson 6 Caesar Cypher

What is a caesar cypher?

It is a method of sending secret messages which was used by the Roman emperor Julius Caesar.

To send a secret message you convert your message (encode it) into what looks like lots of random characters. Only someone who knows your key can tell what you originally wrote! (Actually, this isn't really true - using computers, this kind of cypher would be incredibly easy to crack. Nowadays we use much more complex types of encryption, but the basic principle is still the same).

To encode the message you shift each letter in the message by a number. The number is called the **Key**.

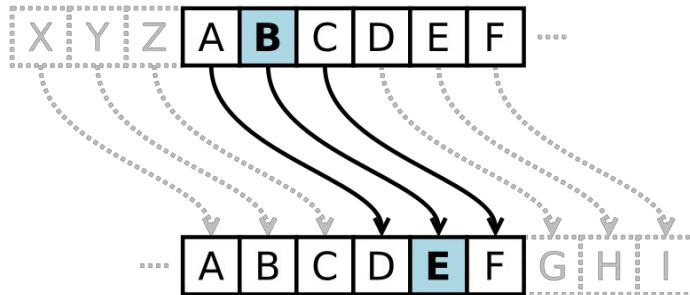
For example if the key was 3 the "new" encoded alphabet would look like the following:

Plain: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Encoded: DEFGHIJKLMNOPQRSTUVWXYZABC

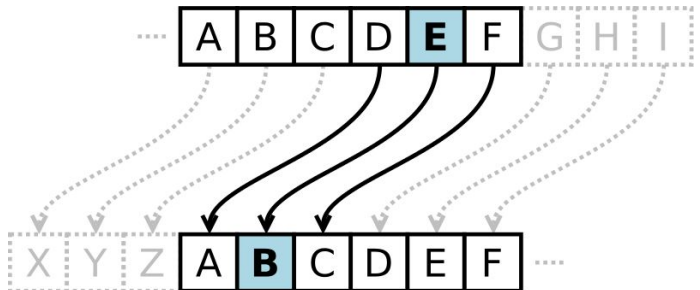
So to "encode" we shift forward by the key (in this example 3) seen below.

Encode b : $(b + \text{cypherKey} = b + 3 = c + 2 = d + 1 = e + 0 = \mathbf{e})$



And to "decode" (find out the meaning of the encoded message) we shift back by the key (in this case 3) - See below:

Decode e : $(e - \text{cypherKey} = e - 3 = d - 2 = c - 1 = b - 0 = \mathbf{b})$



How are we going to use this?

We are going to modify the text input program from lesson 6, so that instead of showing the text that you input it will show the encoded message.

Introducing `char`

`char` is a variable type. Other variable types we've seen include `int` (a number), `String` (one or more words or sentences) and `boolean` (true or false).

What do you think `char` variables are?

Single letters! eg `'a'`, `'b'` etc. But only one letter!

- `'ab'` is wrong - this isn't a `char` at all.
- Only SINGLE quotes. `"a"` is a `String`, whereas `'a'` is a `char`.

Task

Make a caesar cypher encoding program

Do you remember how to show text on the screen?

Make a program that does so using the following:

- `String` variable to contain the message.
- Uses `text()` function to display the message on the screen. (This should be in the `draw()` block)
- Check if the key is greater or equal to `'a'` and less than or equal to `'z'` (`if(key >= 'a' && key <= 'z')`)
 - if so add the `key` to the end of the `String` message variable in the `keyReleased()` block

Use your program from lesson 5 if you get stuck!

To build on this and make a caesar cypher:

- Make a new `String` variable that will contain the encoded (or secret) message.
- Use the `text()` function to display this encoded message in another part of the screen.
- Make a new `int` variable that will contain the cypher key (number of places to shift the message). Set it to some number between 0-25 (eg 4)
- In the `keyReleased()` block, add the encoded key to the encoded message, like below:

```
char encodedLetter = char (key + cypherKey);
encodedMessage += encodedLetter;
```

Check it out! When you enter text you should now see both the original message and the encoded message.

But wait!

Do you see little squares when you press `'z'` or other letters?

We need to check if characters go over the edge of the alphabet (go past 'z'.) If that is the case then we start back at 'a'. Look at the end of the plain alphabet below:

Plain: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Encoded: DEFPGHIJKLMNOPQRSTUVWXYZABC

Let's encode w:

Encode w = w + cypherKey = w +3 = x +2 = y +1 = z +0 = **z**)

Now lets encode x:

Encode x = x + cypherKey = x +3 = y +2 = z +1 = a +0 = **a**)

So if the encoded letter is greater than z - we take the **difference between the encoded letter and z**, **subtract 1** from that, and **count on from a** with that number.

Let's try that by encoding y:

Encode y = y + cypherKey = y +3 = z +2 = a +1 = b +0 = **b**)

How do we do that in code?

```
char encodedLetter = char (key + cypherKey);
//if the encoded letter is past z
if (encodedLetter > 'z') {
    //get the positions past z:      encodedLetter- 'z'
    //minus 1 :                      ( encodedLetter- 'z'-1)
    //go that number past a:        char ('a' +( encodedLetter- 'z'-1)
    encodedLetter = char ('a' +( encodedLetter- 'z'-1));
}
encodedMessage += encodedLetter;
```

Extra tracks:

- Add another **if** statement in the **keyReleased()** block to allow us to add space (' ') to both messages - no encoding needed!
- Show the cypher key on the screen. (using the **text()** function)
- Add mouse clicks to increase and decrease the cypher key (remember the cypher key can be between 0 and 25).

Add buttons that:

- Allow you to change the cypher key

Add decrypting (in a new program - copy the encrypting program across):

- change: (**key** + cypherKey) --> (**key** - cypherKey)
- change the wrapping as well (we need to go the other way):
 - **if** (encodedLetter > 'z') → **if** (encodedLetter < 'a')
 - encodedLetter = **char** ('a' +(encodedLetter- 'z'-1)
 - →
 - encodedLetter = **char** ('z' -('a' - encodedLetter -1)

Add another button that let's you:

- Change from encrypting to decrypting (when changing - clear the message; i.e. set the encodedMessage = "")
- Use a boolean variable for this -eg isDecrypting.