

Caesar Cipher Encrypter and Decrypter

Link to github Repository: github.com

Links to trello board / project management tools: trello.com

Link to a short (2-3 mins) program testing video:

Name of the final completed program:

You MUST provide evidence showing how the problem has been decomposed, how the components have been developed and trialled, and of how they have been assembled and tested to create a final, working outcome.

Authenticity!!



Using generative AI to 'help' is not allowed. NZQA is interested in your knowledge and skills. If you use external resources (like Stack OverFlow, W3C please cite / acknowledge the source).

By submitting this assessment, you declare that the work is your own. This means that you have generated the content yourself and have not looked at any other student's work. If you have used external sources, those have been clearly cited.

Relevant Implication - Functionality

Explain a relevant implication and how you addressed it. Please [watch this video](#) to learn how to do this.

The most important relevant implication is functionality. When making a program we need to make sure it has no glitches or bugs, and that it works as intended. To do this, I have tested for expected, boundary, and unexpected cases. This is important because if a program crashes or gives an incorrect output, the user will become frustrated.

Relevant Implication - Usability

Explain a relevant implication and how you addressed it.

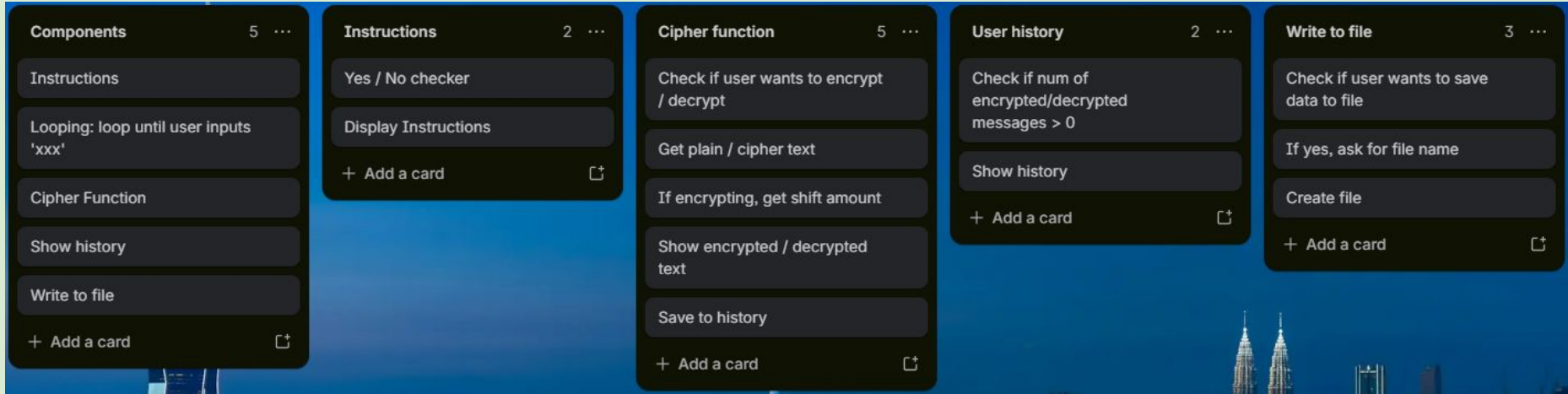
Usability is how easily the end user can accomplish tasks. To achieve this, I made my error messages and instructions easy to understand. Usability is important because if a program is hard to use or understand the user can get frustrated and give up.

Relevant Implication - Sustainability and future-proofing

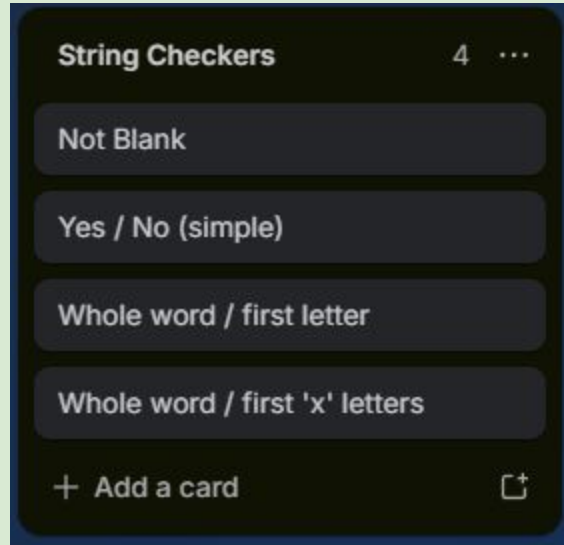
Explain a relevant implication and how you addressed it.

Sustainability and future-proofing is making sure the program can be easily updated or added onto. I have done this by breaking my program down into different components that can be easily updated individually without breaking the rest of the program. This is important because it makes it easier to keep the program working in the future.

Decomposition



Planning - String checker (Project 0)



Test Plan - Not Blank

Data Input	Expected Output (What should happen)
<enter>	Sorry, this can't be blank. Please try again.
abc	Hello abc
12345	Hello 12345

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_02_not_blank_v2.py
Please enter your name:
Sorry, this can't have capital letters, numbers, special characters, or be blank. Please try again.

Please enter your name: 12345
Sorry, this can't have capital letters, numbers, special characters, or be blank. Please try again.

Please enter your name: abc
Hello abc
```

Test Plan - Not Blank

Data Input	Expected Output (What should happen)
!@#\$%	Error
%g0)hfG&^98	Error
name	Hello name

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_02_not_blank_v2.py
Please enter your name: !@#$%
Sorry, this can't have capital letters, numbers, special characters, or be blank. Please try again.

Please enter your name: %g0)hfG&^98
Sorry, this can't have capital letters, numbers, special characters, or be blank. Please try again.

Please enter your name: name
Hello name
```

Test Plan - Yes / No Simple

Data Input	Expected Output (What should happen)
yes	Program continues
Y	Program continues
YeS	Program continues

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C03_yes_no_simple.py
Do you want to read the instructions? yes
You chose yes

Do you want to read the instructions? Y
You chose yes

Do you want to read the instructions? YeS
You chose yes
```

Test Plan - Yes / No Simple

Data Input	Expected Output (What should happen)
No	Program continues
N	Program continues
Maybe	Please enter either “Yes” or “No”

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C03_yes_no_simple.py
Do you want to read the instructions? No
You chose no

Do you want to read the instructions? N
You chose no

Do you want to read the instructions? Maybe
Please enter yes (y) or no (n).
```

Test Plan - Whole word / first letter

Data Input	Expected Output (What should happen)
Maybe Yes	Sorry, please enter 'yes' or 'no' You chose 'yes'
simple H	Sorry, please choose easy, medium, or hard You chose 'hard'
easy	You chose 'easy'

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_04_string_check_v1.py
Do you like coffee? Maybe
Please choose an option from ['yes', 'no']
Do you like coffee? Yes
You chose yes
Choose a level: simple
Please choose an option from ['easy', 'medium', 'hard']
Choose a level: H
You chose hard
```

Test Plan - Whole word / first 'x' letters

Data Input	Expected Output (What should happen)
Maybe	Please choose an option from ['yes', 'no']
Yes	You chose yes
coins	Please choose an option from ['cash', 'credit']
c	Please choose an option from ['cash', 'credit']
ca	You chose cash
CREDIT	You chose credit

Test Plan - Whole word / first 'x' letters

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrighards\Programming\2026_Project_Zero\C_04_string_check_v2.py
Do you like coffee? Maybe
Please choose an option from ['yes', 'no']
Do you like coffee? Yes
You chose yes
Payment method: coins
Please choose an option from ['cash', 'credit']
Payment method: c
Please choose an option from ['cash', 'credit']
Payment method: ca
You chose cash
```

```
Payment method: CREDIT
You chose credit
```

Trialing - String Checker

Trial 1 - Simple yes/no checker (only checks for yes/no)

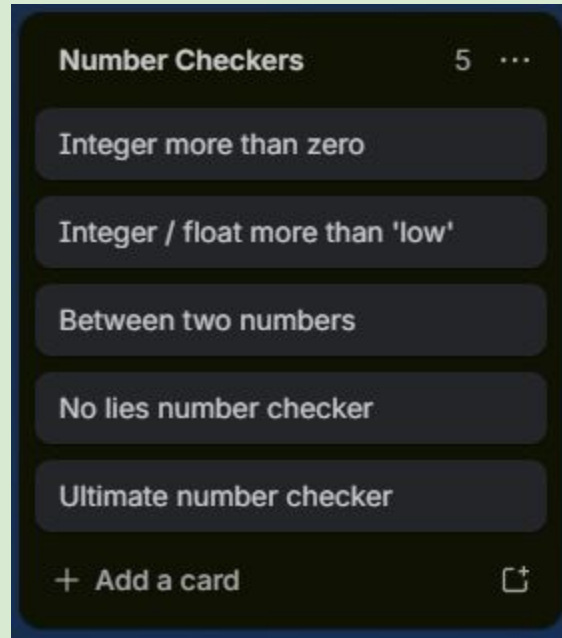
Trial 2 - Checks users enter a valid option from a list (first letter / whole word, returns the whole word)

Trial 3 - Checks users enter a valid option from a list (variable letters / whole word). Might be overkill for most situations.

Trial 4 - Similar to Trial 2 but defaults to yes / no responses

Used component from trial 2, simplest version while still having custom lists

Planning - Number Checkers (Project 0)



Test Plan - Float or integer more than zero

Data Input	Expected Output (What should happen)
XLII	Please enter a number.
0	Please enter a number more than 0
.1	Program continues

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_05_number_checker_v1.py
```

```
Please enter a number more than 0: XLII
Oops - please enter a number more than zero.
Please enter a number more than 0: 0
Oops - please enter a number more than zero.
Please enter a number more than 0: .1
Thanks. You chose 0.1
```

Test Plan - Float or integer more than zero

Data Input	Expected Output (What should happen)
XLII	Please enter an integer (ie: a number which does not have a decimal part).
0.5	Please enter an integer (ie: a number which does not have a decimal part).
0	Please enter an integer that is more than 0
1	Program continues
<enter>	You have chosen infinite mode.

Test Plan - Float or integer more than zero

Actual Output (What actually happened)

```
Please enter an integer more than 0: | XLII
Oops - please enter an integer more than zero.
Please enter an integer more than 0: 0.5
Oops - please enter an integer more than zero.
Please enter an integer more than 0: 0
Oops - please enter an integer more than zero.
Please enter an integer more than 0: 1
Thanks. You chose 1
```

Test Plan - Int Check Between

Data Input	Expected Output (What should happen)
XLII	Please enter an integer (ie: a number which does not have a decimal part).
1.5	Please enter an integer (ie: a number which does not have a decimal part).
0	Please enter an integer (ie: a number which does not have a decimal part).
1	Program continues
2	Program continues
9	Program continues
10	Program continues
11	Please enter an integer that is less than or equal to 10

Test Plan - Int Check Between

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_05_integer_checker_between_v2.
```

```
Choose a number: XLII
```

```
Oops - please enter an integer between 1 and 10.
```

```
Choose a number: 1.5
```

```
Oops - please enter an integer between 1 and 10.
```

```
Choose a number: 0
```

```
Oops - please enter an integer between 1 and 10.
```

```
Choose a number: 1
```

```
You chose 1
```

```
Choose a number: 2 Choose a number: 9 Choose a number: 10
```

```
You chose 2 You chose 9 You chose 10
```

```
Choose a number: 11
```

```
Oops - please enter an integer between 1 and 10.
```

```
Choose a number:
```

Test Plan - No Lies Number Checker

Data Input	Expected Output (What should happen)
A, XLII	Please enter an integer between 12 and 120
A, 12.5	Please enter an integer between 12 and 120
A, 11	A is too young
B, 12	B has bought a ticket
C, 13	C has bought a ticket
D, 119	D has bought a ticket
E, 120	E has bought a ticket
F, 121	F is too old

Test Plan - No Lies Number Checker

Actual Output (What actually happened)

```
C:\Python39\python.exe W:\12\isaacrichards\Programming\2026_Project_Zero\C_06_no_lies_integer_checker.py

Name: A
Age: XLII
Oops - please enter an integer more than zero.
Age: 12.5
Oops - please enter an integer more than zero.
Age: 11
A is too young

Name: B
Age: 12
B bought a ticket

Name: C
Age: 13
C bought a ticket

Name: D
Age: 119
D bought a ticket

Name: E
Age: 120
E bought a ticket

Name: F
Age: 121
F is too old
```

Trialing - Number Checkers

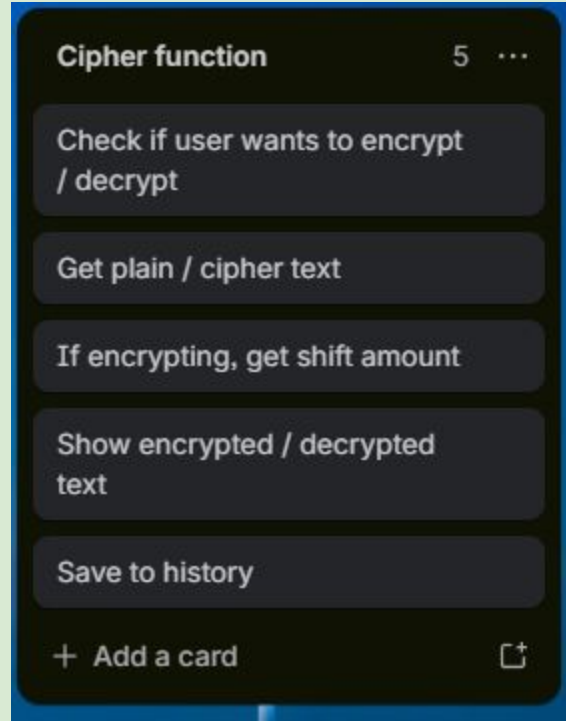
Trial 1 - Checks for integers / floats that are more than zero with an optional, customisable exit code

Trial 2 - Checks users enter an integer between two values

Trial 3 - No lies integer checker - checks for an integer and if the number is invalid, main routine loop goes back to the start

Most useful version would be trial 2. We need to check users pick a number between 0 and 26.

Planning - Cipher Function



Test Plan - Cipher Function v1

Data Input	Expected Output (What should happen)
Encrypt, 5, Hello!	Hjqqt!
Encrypt, 19, Message	Mxlltzx
Decrypt, 5, Hjqqt!	Hello!
Decrypt, 19, Mxlltzx	Message
xxx	Program Continues

Test Plan - Cipher Function v1

Actual Output (What actually happened)

```
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/python.exe w:/12/isaacrichards/Programming/2026-91896_91897_Caesar_Cipher/C_01_cipher_func_v1.py
Encrypt or Decrypt? e

Plain text: Hello!

Shift amount: 5

Encrypted Message: Hjqqt!

Encrypt or Decrypt? e

Plain text: Message

Shift amount: 19

Encrypted Message: Mxlltzx

PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/python.exe w:/12/isaacrichards/Programming/2026-91896_91897_Caesar_Cipher/C_01_cipher_func_v1.py
Encrypt or Decrypt? d

Cipher Text: Hjqqt!

Shift amount: 5

Decrypted Message: Hello!

Encrypt or Decrypt? d

Cipher Text: Mxlltzx

Shift amount: 19

Decrypted Message: Message

Encrypt or Decrypt? xxx

PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher>
```

Test Plan - Cipher Function v2

Data Input	Expected Output (What should happen)
Encrypt, 5, Hello!	Mjqqt!
Encrypt, 19, Message	Fxlltzx
Decrypt, Mjqqt!	List of possible options
Decrypt, Fxlltzx	List of possible options
xxx	Program Continues

Test Plan - Cipher Function v2

Actual Output (What actually happened)

```
PS W:\12\isaacr\richards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/python.exe w:/12/isaacr\richards\Programming/
2026-91896_91897_Caesar_Cipher/C_01_cipher_func_v2.py
Encrypt or Decrypt? e

Plain text: Hello!

Shift amount: 5

Encrypted Text: Mjqqt!

Encrypt or Decrypt? e

Plain text: Message

Shift amount: 19

Encrypted Text: Fxlltzx

Encrypt or Decrypt? d

Cipher Text: Mjqqt!

Shift 1: Lipps!
Shift 2: Khoor!
Shift 3: Jgnnq!
Shift 4: Ivmvp!
Shift 5: Hello!
Shift 6: GdKkn!
Shift 7: Fcjjn!
Shift 8: Ebiil!
Shift 9: Dahhk!
Shift 10: Czggj!
Shift 11: Byffl!
Shift 12: Axeeh!
Shift 13: Zuddg!
Shift 14: Yvccf!
Shift 15: Xubbe!
Shift 16: Wtaad!
Shift 17: Vszzc!
Shift 18: Uryyb!
Shift 19: Tapxa!
Shift 20: Spwaz!
Shift 21: Rvvvy!
Shift 22: Qnuux!
Shift 23: Pmttw!
Shift 24: Olssv!
Shift 25: Nkrnu!

Shift 1: Ewkkisyw
Shift 2: Dvjrrxv
Shift 3: Cuiiqwu
Shift 4: Bthhpvt
Shift 5: Asggous
Shift 6: Zrffntr
Shift 7: Yqeemsq
Shift 8: Xpddlrp
Shift 9: Wocckqo
Shift 10: Vnbbjpn
Shift 11: Umaaiom
Shift 12: Tlzzhnl
Shift 13: SkyygmK
Shift 14: Rjxxflj
Shift 15: Qlwweki
Shift 16: Phvvdjh
Shift 17: Oguucig
Shift 18: Nfttbhf
Shift 19: Message
Shift 20: Ldrrzfd
Shift 21: Kcqyqec
Shift 22: Jbppxdb
Shift 23: Iaoowca
Shift 24: Hznnvbz
Shift 25: Gyumuay

Encrypt or Decrypt? xxx

PS W:\12\isaacr\richards\Programming\2026-91896_91897_Caesar_Cipher> |
```

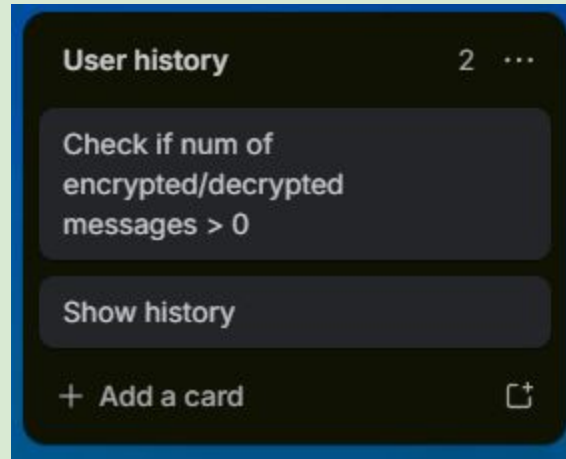
Trialing - Cipher Function

Trial 1 - Use list to encrypt / decrypt message

Trial 2 - Use ASCII to encrypt / decrypt message

I will use Trial 2. It doesn't require a cipher key to decrypt message.

Planning - User History



Test Plan - User History v1

Data Input

```
un_encrypted_messages = ['Hello', 'Creative Message']
encrypted_messages = ['Zwddg', 'Fuhdwlyh Phvvdjh']
un_decrypted_messages = ['Aopz pz h tlzzhnl', 'Gybnc kxn cdepp']
decrypted_messages_lists = ['Znoy oy g skyygmK, Ymnx nx f rjxxflj, Xlmw mw e qiwweki, Wklv lv d phvvdjh,
```

Expected Output

Plain Text	Encrypted Message
Hello	Zwddg
Creative Message	Fuhdwlyh Phvvdjh

Cipher Text	Decrypted Messages
Aopz pz h tlzzhnl	List decrypted messages
Znoy oy g skyygmK	List decrypted messages

Test Plan - User History v1

Actual Output

```
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/python.exe w:/12/isaacrichards/Programming/2026-91896_91897_Caesar_Cipher/C_02_user_history_v1.py
```

```
Plain Text Encrypted Message
Hello                Zwddg
Creative Message     FuhduLyh Phvvdjh
```

Cipher Text

Decrypted Messages

```
Aopz pz h tizzhnl Znqy oy g skyygmK, Ymnx nx f rjxxflj, Xlmw mw e qiwweki, Wklv lv d phvvdjh, Vjku ku c oguucig, Uijt jt b nfttbhf, This is a message, Sghr hr z ldrzfd, Rfga ga y kcaqyec, Qefp fp x jpppxdb, Pdeo eo w iaoowca, Ocdn dn v hznvzbz,
zj r dvjrrxv, Jxyi yi q cuiliqw, Iwxh xh p bthhpvt, Hwgg wg o asggous, Guvf vf n zrffntr, Ftue ue m ygeemsg, Estd td l xpddlrp, Drsc sc k wocckgo, Cqrb rb j vnbbjpn, Bpqa qa i umaaiom
Gybnc knx cdepp          Fxamb jwm bcdoo, Ewzla ivl abcnm, Dyykz huk zabmm, Cuxjy gtj yzall, Btwix fsi xyzkk, Asvhw erh woyvj, Zrugv dag vxii, Yqtfu cpf uvwhh, Xpset boe tuvvg, Words and stuff, Vnqcr zm
jmy n viy nopaa, Qilxm uhx mnozz, Phkw l tgu lmyy, Ogjvk sfv klmox, Nfiuj reu jklmw, Mehti qdt ijkvv, Ldgsh pcs hijuu, Kcfrg obr ghitt, Jbeqf naq fghss, Iadpe mzp efgrr, Hzcod lyo defqq
```

```
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> █
```

Test Plan - User History v2

Data Input

```
un_encrypted_messages = ['Hello', 'Creative Message']
encrypted_messages = ['Zwddg', 'Fuhdwlyh Phvvdjh']
un_decrypted_messages = ['Aopz pz h tlzzhnl', 'Gybnc kxn cdepp']
decrypted_messages_lists = ['Znoy oy g skyygmK, Ymnx nx f rjxxflj, Xlmm mw e qiwweki, Wklv lv d phvvdjh,
```

Expected Output

---Encrypted Messages---

Hello --> Zwddg

Creative Message --> Fuhdwlyh Phvvdjh

---Decrypted Messages---

Aopz pz h tlzzhnl:

List decrypted messages

Gybnc kxn cdepp:

List decrypted messages

Test Plan - User History v2

Actual Output (What actually happened)

```
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/python.exe w:/12/isaacrichards/Programming/2026-91896_91897_Cipher/C_02_user_history_v2.py
---Encrypted Messages---

Hello --> Zwddg

Creative Message --> Fuhdwlyh Phvvdjh

---Decrypted Messages---

Aopz pz h tlzzhnl:

Znoy oy g skyygmK, Ymnx rx f rjxxflj, Xlmw mw e qiwweki, Wklv lv d phvvdjh, Vjku ku c oguucig, Uijt jt b nfttbhf, This is a mes
zfd, Rfgq gq y kcqqyec, Qefp fp x jbppxdb, Pdeo eo w iaooowca, Ocdn dn v hznnvz, Nbcm cm u gymmuay, Mabl bl t fxlltzx, Lzak ak
dvjrxv, Jxyi yi q cuiqwu, Iwxh xh p bthhpvt, Hwgg wg o asggous, Gufv vf n zrffntr, Ftue ue m yqeemsg, Estd td l xpddlrp, Drs
rb j vnbbjpn, Bpqa qa i umaaiom

Gybnc kxn cdepp:

Fxamb jwm bcdoo, Ewzla ivl abcn, Dvykz huk zabmm, Cuxjy gtj yzall, Btwix fsi xyzkk, Asvhw erh wxyjj, Zrugv dqg vwxii, Yqtfu cp
uvgg, Words and stuff, Vnqcr zmc rstee, Umpbq ylb qrsdd, Tloap xka qqrcc, Sknzo wjz opqbb, Rjmyn viy nopaa, Qilxm uhx mnozz, Ph
sfv klmxx, Nfiuj reu jklww, Mehti qdt ijkvv, Ldgsh pcs hijuu, Kcfrg obr ghitt, Jbeqf naq fghss, Iadpe mzp efgrr, Hzcod lyo def

PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> |
```

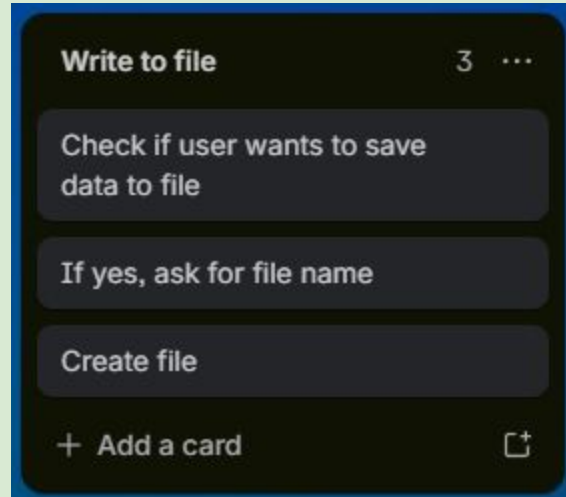
Trialing - User History

Trial 1 - Use pandas to put data into table

Trial 2 - Use for loop to print data

I will use trial 1, easier to read than pandas

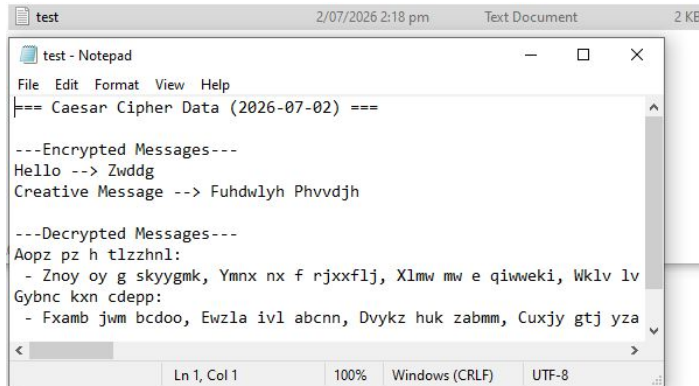
Planning - Write to file



Test Plan - Write to file

Data Input	Expected Output (What should happen)
y, test	Text file named 'test' created, has caesar cipher data
n	Program Continues

Actual Output (What actually happened)



The screenshot shows a Notepad window titled 'test' with the following content:

```
File Edit Format View Help
=== Caesar Cipher Data (2026-07-02) ===

---Encrypted Messages---
Hello --> Zwddg
Creative Message --> Fuhdwlyh Phvvdjh

---Decrypted Messages---
Aopz pz h tlzzhnl:
- Znpy oy g skyygm, Ymnx nx f rjxxflj, Xlmw mw e qiwweki, Wklv lv
Gybnc kxn cdepp:
- Fxamb jwm bcdoo, Ewzla ivl abcn, Dvykz huk zabmm, Cuxjy gtj yza
```

```
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher> & C:/Python39/pyth
Would you like to save your data to a file? n
PS W:\12\isaacrichards\Programming\2026-91896_91897_Caesar_Cipher>
```

Version Control

- B_01_caesar_cipher_v1
- B_01_caesar_cipher_v2
- C_01_cipher_func_v1
- C_01_cipher_func_v2
- C_02_user_history_v1
- C_02_user_history_v2
- C_03_int_check
- C_04_string_check_v1
- E_02_write_to_file

B_01_caesar_cipher - v1 had instructions, encryption / decryption. v2 added history and write to file.

C_01_cipher_func - v1 used a list of the alphabet to encrypt / decrypt the message. v2 uses ASCII codes instead.

C_02_user_history - v1 used pandas to save history, but it didn't print nicely. v2 uses lists to save data, and prints in a way that is much easier to read.

C_03_int_check - No need for multiple versions, very simple function.

C_04_string_check - Other string check versions were excessive for what I needed, so I took only v1.

E_02_write_to_file - Once again, a simple function with no need for multiple versions.

Improvements & Refinements

Refinement: Results now show shift amount

Before	After
<pre>---Encrypted Messages--- Test --> Yjxy</pre>	<pre>---Encrypted Messages--- Plain text: Test Shift amount: 5 CIPHER text: Yjxy</pre>

This lets users easily see what the shift amount was for each of their encrypted messages, rather than forcing them to remember.

Improvements & Refinements

Refinement: Improved instructions

Before	After
<pre data-bbox="369 341 604 503">---Instructions--- Choose a mode Enter the text xxx to exit</pre>	<pre data-bbox="954 341 1707 582">---Instructions--- - First, choose either encrypt or decrypt. - Next, enter the plain / cipher text. - If you have chosen encrypt, choose an amount to shift. - You will then see the encrypted / decrypted text. - To exit the code, choose xxx instead of encrypt / decrypt.</pre>

New instructions are much more clear and less confusing than old instructions.

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
	Please choose an option from ['yes', 'no']	Would you like to see the instructions? Please choose an option from ['yes', 'no'] Would you like to see the instructions? █
maybe	Please choose an option from ['yes', 'no']	Would you like to see the instructions? maybe Please choose an option from ['yes', 'no'] Would you like to see the instructions? █
ABCD	Please choose an option from ['yes', 'no']	Would you like to see the instructions? ABCD Please choose an option from ['yes', 'no'] Would you like to see the instructions? █
12345	Please choose an option from ['yes', 'no']	Would you like to see the instructions? 12345 Please choose an option from ['yes', 'no'] Would you like to see the instructions? █
!@#\$	Please choose an option from ['yes', 'no']	Would you like to see the instructions? !@#\$ Please choose an option from ['yes', 'no'] Would you like to see the instructions? █
Yes	Program Continues	Would you like to see the instructions? Yes ---Instructions---
y	Program Continues	Would you like to see the instructions? y ---Instructions---

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
No	Program Continues	<pre>Would you like to see the instructions? No Encrypt or Decrypt? █</pre>
N	Program Continues	<pre>Would you like to see the instructions? N Encrypt or Decrypt? █</pre>
n	Program Continues	<pre>Would you like to see the instructions? n Encrypt or Decrypt? █</pre>
no	Program Continues	<pre>Would you like to see the instructions? no Encrypt or Decrypt? █</pre>
yes	Program Continues	<pre>Would you like to see the instructions? yes ---Instructions---</pre>
Y	Program Continues	<pre>Would you like to see the instructions? Y ---Instructions---</pre>

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
	Please choose an option from ['encrypt', 'decrypt', 'xxx']	Encrypt or Decrypt? Please choose an option from ['encrypt', 'decrypt', 'xxx'] Encrypt or Decrypt? █
idk	Please choose an option from ['encrypt', 'decrypt', 'xxx']	Encrypt or Decrypt? idk Please choose an option from ['encrypt', 'decrypt', 'xxx'] Encrypt or Decrypt? █
ABCD	Please choose an option from ['encrypt', 'decrypt', 'xxx']	Encrypt or Decrypt? ABCD Please choose an option from ['encrypt', 'decrypt', 'xxx'] Encrypt or Decrypt? █
12345	Please choose an option from ['encrypt', 'decrypt', 'xxx']	Encrypt or Decrypt? 12345 Please choose an option from ['encrypt', 'decrypt', 'xxx'] Encrypt or Decrypt? █
!@#\$	Please choose an option from ['encrypt', 'decrypt', 'xxx']	Encrypt or Decrypt? !@#\$ Please choose an option from ['encrypt', 'decrypt', 'xxx'] Encrypt or Decrypt? █
Encrypt	Program Continues	Encrypt or Decrypt? Encrypt Plain text: █
E	Program Continues	Encrypt or Decrypt? E Plain text: █

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
ENCRYPT	Program Continues	Encrypt or Decrypt? ENCRYPT Plain text: █
encrypt	Program Continues	Encrypt or Decrypt? encrypt Plain text: █
e	Program Continues	Encrypt or Decrypt? e Plain text: █
Decrypt	Program Continues	Encrypt or Decrypt? Decrypt Cipher Text: █
DECRYPT	Program Continues	Encrypt or Decrypt? DECRYPT Cipher Text: █
decrypt	Program Continues	Encrypt or Decrypt? decrypt Cipher Text: █
d	Program Continues	Encrypt or Decrypt? d Cipher Text: █
D	Program Continues	Encrypt or Decrypt? D Cipher Text: █

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
xxx	Program Continues	<pre>Encrypt or Decrypt? xxx ---Encrypted Messages--- Plain text: message Shift amount: 5 Cipher text: rjxxflj ---Decrypted Messages--- rjxxflj: - qiwweki, phvvdjh, oguucig, nfttbhf, message, ldrzfd, kcqy</pre>

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
message, 1	Encrypted Text: nfttbhf	Plain text: message Shift amount: 1 Encrypted Text: nfttbhf
message, 25	Encrypted Text: ldrzfd	Plain text: message Shift amount: 25 Encrypted Text: ldrzfd
message, 15	Encrypted Text: bthhpvt	Plain text: message Shift amount: 15 Encrypted Text: bthhpvt
message, 0	Please enter an integer between 0 and 26.	Plain text: message Shift amount: 0 Please enter an integer between 0 and 26. Shift amount:
message, 26	Please enter an integer between 0 and 26.	Plain text: message Shift amount: 26 Please enter an integer between 0 and 26. Shift amount:
message, !@#	Please enter an integer between 0 and 26.	Plain text: message Shift amount: !@# Please enter an integer between 0 and 26. Shift amount:
message, ABC	Please enter an integer between 0 and 26.	Plain text: message Shift amount: ABC Please enter an integer between 0 and 26. Shift amount:
message, abc	Please enter an integer between 0 and 26.	Plain text: message Shift amount: abc Please enter an integer between 0 and 26. Shift amount:

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
y	Program Continues	<pre>Would you like to save your data to a file? y File Name: █</pre>
Y	Program Continues	<pre>Would you like to save your data to a file? Y File Name: █</pre>
Yes	Program Continues	<pre>Would you like to save your data to a file? Yes File Name: █</pre>
YES	Program Continues	<pre>Would you like to save your data to a file? YES File Name: █</pre>
N	Program Ends	<pre>Would you like to save your data to a file? N PS W:\12\isaacrichards\Programming\91896_isaacrichards> █</pre>
n	Program Ends	<pre>Would you like to save your data to a file? n PS W:\12\isaacrichards\Programming\91896_isaacrichards> █</pre>
no	Program Ends	<pre>Would you like to save your data to a file? no PS W:\12\isaacrichards\Programming\91896_isaacrichards> █</pre>
NO	Program Ends	<pre>Would you like to save your data to a file? NO PS W:\12\isaacrichards\Programming\91896_isaacrichards> █</pre>

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
maybe	Please choose an option from ['yes', 'no']	<pre>Would you like to save your data to a file? maybe Please choose an option from ['yes', 'no'] Would you like to save your data to a file? █</pre>
123	Please choose an option from ['yes', 'no']	<pre>Would you like to save your data to a file? 123 Please choose an option from ['yes', 'no'] Would you like to save your data to a file? █</pre>
!@#	Please choose an option from ['yes', 'no']	<pre>Would you like to save your data to a file? !@# Please choose an option from ['yes', 'no'] Would you like to save your data to a file? █</pre>

Comprehensive Testing Evidence - Final Program Version

Data Input	Expected Output	Actual Output
cipher_data	File created, named 'cipher_data'	 cipher_data 23/07/2026 2:27 pm Text Document 2 KB
CIPHER_DATA	File created, named 'CIPHER_DATA'	 CIPHER_DATA 23/07/2026 2:28 pm Text Document 2 KB
Cipher Data	File created, named 'Cipher Data'	 Cipher Data 23/07/2026 2:29 pm Text Document 2 KB
12345	File created, named '12345'	 12345 23/07/2026 2:29 pm Text Document 2 KB
!@#\$%	File created, named '!@#\$%'	 !@#\$% 23/07/2026 2:30 pm Text Document 2 KB
hi.html	File created, named 'hi.html'	 hi.html 23/07/2026 2:30 pm Text Document 2 KB

Discussion

I found that planning out what components I needed to create with a trello board gave me a clear goal to work towards. I didn't find trialling very useful. I found that making multiple pieces of code for one task was inefficient and time consuming. Testing helped me find and fix small errors I had made while coding. I didn't manage to get much feedback on my program, so it may be limited in that way. Next time I will try to get more feedback from more sources.