

Code Review 09 – Tkinter

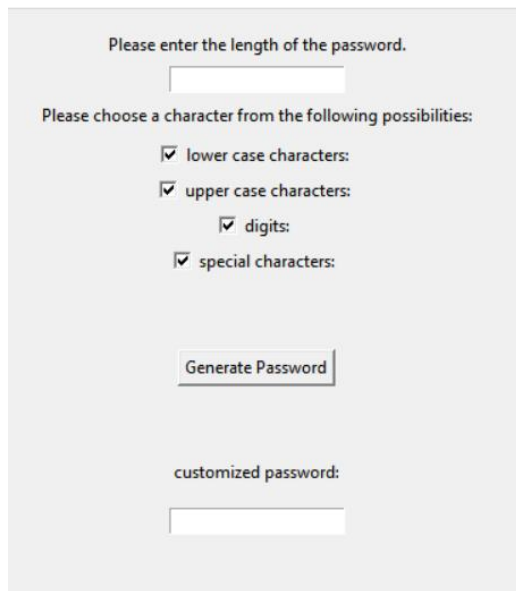
Autor: Jana Hofmann

Matrikelnummer: 8033684

Datum: 07.02.2026

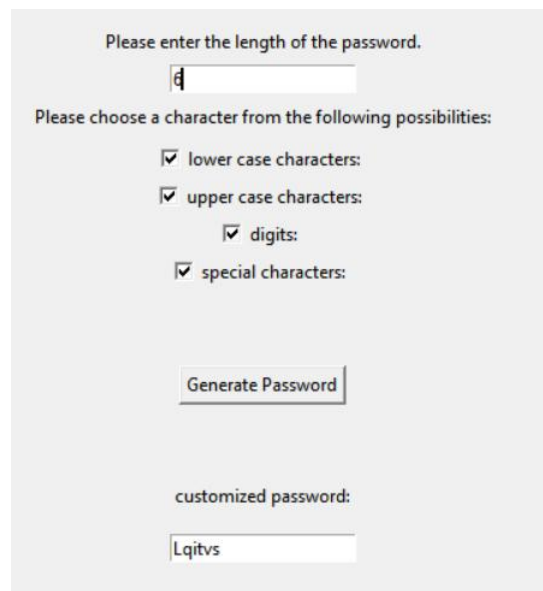
Aufgabe 1) Tkinter: password Generator

Erstellen eines Passwort Generator mit einer externen graphischen Benutzeroberfläche (GUI) der es dem Benutzer ermöglicht ein zufälliges Passwort mit verschiedenen Zeichen (lower-case, upper_case, digits, special_characters) zu generieren. Das Programm besteht aus drei Modulen, der main.py, die den Einstiegspunkt des Programms darstellt, weiterhin dem Passwort Generator pw_generator_1 und der GUI gui_pw_gen-1. Beim Ausführen des Programms über die main erscheint ein Fenster, in dem man die Länge des Passworts sowie die verschiedenen Charakter auswählen kann. Das Passwort wird über einen Button generiert und unten im Fenster ausgegeben.



The screenshot shows a Tkinter window titled "Please enter the length of the password." with a text input field. Below it, the text "Please choose a character from the following possibilities:" is followed by four checked checkboxes: "lower case characters:", "upper case characters:", "digits:", and "special characters:". A "Generate Password" button is located below the checkboxes. At the bottom, there is a label "customized password:" and an empty text input field.

Leeres Fenster des Passwort Generators



The screenshot shows the same Tkinter window as the previous one, but with the text input field containing the number "5". The "Generate Password" button has been clicked, and the generated password "Lqitvs" is now displayed in the "customized password:" text input field at the bottom.

Generieren eines zufälligen Passworts mit allen möglichen Zeichen und einer Länge von 5 Zeichen

Please enter the length of the password.

Please choose a character from the following possibilities:

☐ lower case characters:
☒ upper case characters:
☒ digits:
☒ special characters:

customized password:

Ausschluss der Kleinbuchstaben

Please enter the length of the password.

Please choose a character from the following possibilities:

☒ lower case characters:
☐ upper case characters:
☒ digits:
☒ special characters:

customized password:

Ausschluss der Großbuchstaben

Please enter the length of the password.

Please choose a character from the following possibilities:

☒ lower case characters:
☒ upper case characters:
☐ digits:
☒ special characters:

customized password:

Ausschluss der Zahlen

Please enter the length of the password.

Please choose a character from the following possibilities:

☒ lower case characters:
☒ upper case characters:
☒ digits:
☐ special characters:

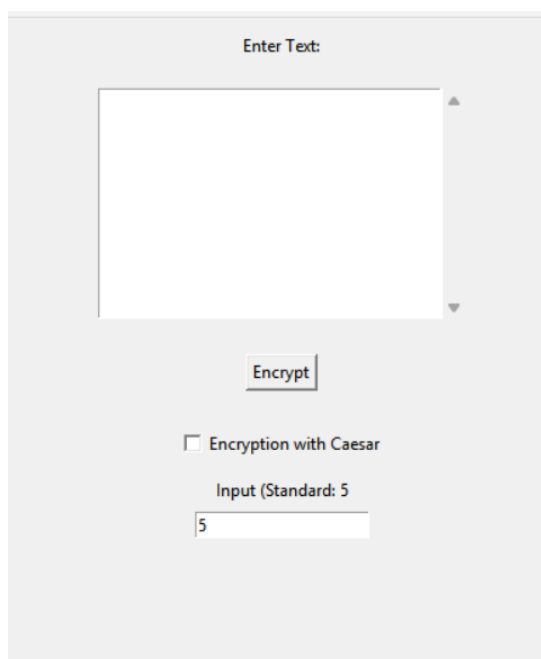
customized password:

Ausschluss der Sonderzeichen

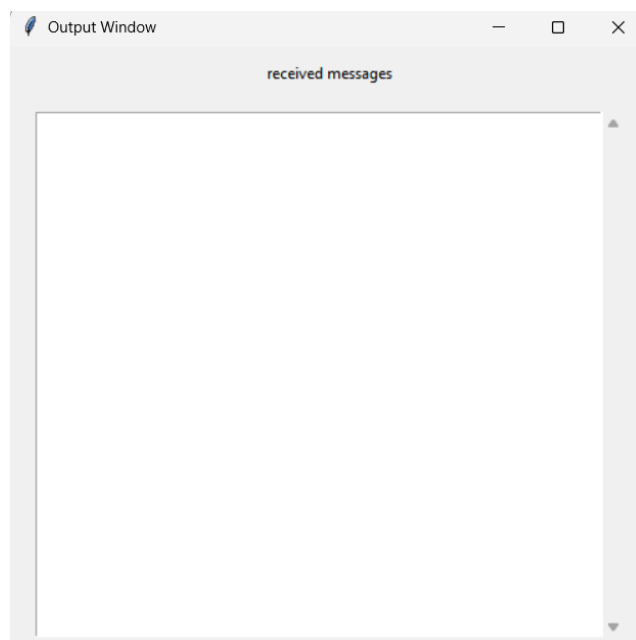
Aufgabe 2) Tkinter: verschlüsselte Fenster

Erstellen eines Python Programms mit Tkinter und Ceasar-Verschlüsselung, das es dem Benutzer ermöglicht Texte/ Wörter automatisch zu verschlüsseln und ausgeben zu lassen. Das Programm besteht aus drei Modulen, einer `main.py`, die als Einstiegspunkt des Programms dient und vom Benutzer ausgeführt werden muss, weiterhin der GUI `encrypted_gui_2.py` und der Ceasar-Verschlüsselung `caesar_cypher_2.py`.

Beim Ausführen des Programms über `main` öffnen sich zwei Fenster, ein Eingabe Fenster (Input-Window) auf dem der Benutzer seinen text eingeben kann und auswählen kann ob der Text mit ceasar verschlüsselt und in wie der shuffle sein soll. Gleichzeitig öffnet sich das Ausgabe Fenster (Output Window) auf dem der potenziell verschlüsselte Text ausgegeben wird.



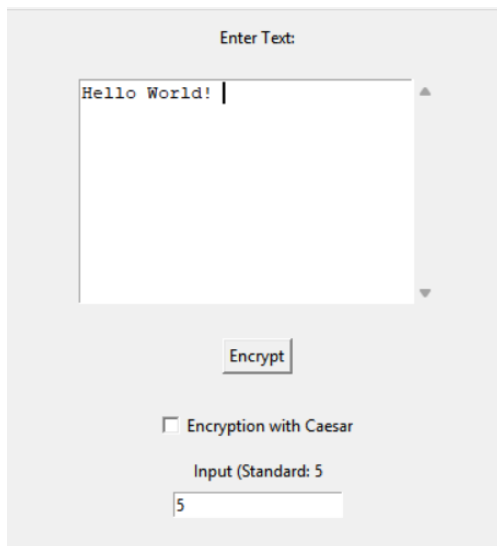
Eingabe Fenster leer



Ausgabe Fenster leer

Tests

Ohne Verschlüsselung



Enter Text:

Hello World! |

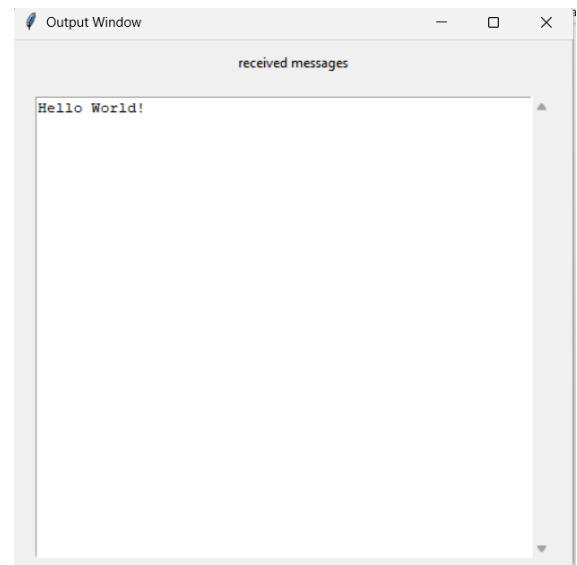
Encrypt

☐ Encryption with Caesar

Input (Standard: 5)

5

Eingabe Fenster mit einem Satz
ohne Verschlüsselung



Output Window

received messages

Hello World!

Ausgabe Fenster mit dem
unverschlüsselten Satz

Tests mit Verschlüsselung



Enter Text:

Hello World! |

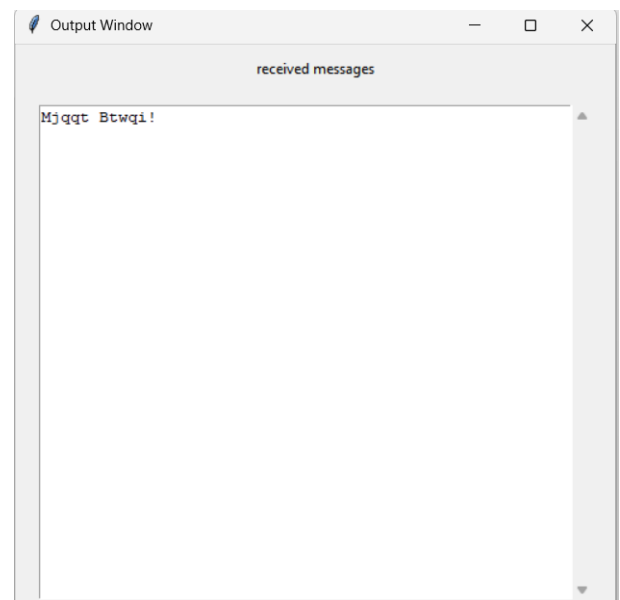
Encrypt

☒ Encryption with Caesar

Input (Standard: 5)

5

Eingabe Fenster mit einem Satz mit
Verschlüsselung (shuffle = 5)



Output Window

received messages

Mjqqt Btwqi!

Ausgabe Fenster mit
verschlüsseltem Satz