

EL CIFRADO CÉSAR: ENCRIPTANDO UN MENSAJE

Te mostraremos cuatro programas simples para presentar algunos aspectos del procesamiento de cadenas en Python. Son intencionalmente simples, pero los problemas de laboratorio serán significativamente más complicados.

Este cifrado fue (probablemente) inventado y utilizado por Cayo Julio César y sus tropas durante las Guerras Galas. La idea es bastante simple: cada letra del mensaje se reemplaza por su consecuente más cercano (*A* se convierte en *B*, *B* se convierte en *C*, y así sucesivamente). La única excepción es *Z*, la cual se convierte en *A*.

El programa es una implementación muy simple (pero funcional) del algoritmo.

Se ha escrito utilizando los siguientes supuestos:

- Solo acepta letras latinas (nota: los romanos no usaban espacios en blanco ni dígitos).
- Todas las letras del mensaje están en mayúsculas (nota: los romanos solo conocían las mayúsculas).

Veamos el código:

- La línea 02: pide al usuario que ingrese un mensaje (sin cifrar) de una línea.
- La línea 03: prepara una cadena para el mensaje cifrado (esta vacía por ahora).
- La línea 04: inicia la iteración a través del mensaje.
- La línea 05: si el caracter actual no es alfabético...
- La línea 06: ...ignoralo.
- La línea 07: convierta la letra a mayúsculas (es preferible hacerlo a ciegas, en lugar de verificar si es necesario o no).
- La línea 08: obtén el código de la letra e increméntalo en uno.
- La línea 09: si el código resultante ha "dejado" el alfabeto latino (si es mayor que el código de la *Z*)...
- La línea 10: ... cámbialo al código de la *A*.
- La línea 11: agrega el carácter recibido al final del mensaje cifrado.
- La línea 13: imprime el cifrado.

El código, alimentado con este mensaje:

```
AVE CAESAR
```

Da como salida:

```
BWFDBFTBS
```

```
# Cifrado César

text = input("Ingresa tu mensaje: ")
cifrado = ''
for char in text:
    if not char.isalpha():
        continue
    char = char.upper()
    code = ord(char) + 1
    if code > ord('Z'):
        code = ord('A')
    cifrado += chr(code)
print(cifrado)
```

EL CIFRADO CÉSAR: DESCIFRANDO UN MENSAJE

La operación inversa ahora debería ser clara para ti: solo presentamos el código tal como está, sin ninguna explicación.

Observa el código en el editor. Comprueba cuidadosamente si funciona. Usa el criptograma del programa anterior.

```
# Cifrado César - descifrar un mensaje
cifrado = input('Ingresa tu criptograma: ')
text = ''
for char in cifrado:
    if not char.isalpha():
        continue
    char = char.upper()
    code = ord(char) - 1
    if code < ord('A'):
        code = ord('Z')
    text += chr(code)
print(text)
```

ESCENARIO

Ya estás familiarizado con el cifrado César, y es por eso que queremos que mejores el código que te mostramos recientemente.

El cifrado César original cambia cada caracter por otro a se convierte en b , z se convierte en a , y así sucesivamente. Hagámoslo un poco más difícil y permitamos que el valor desplazado provenga del rango 1 al 25.

Además, dejemos que el código conserve las mayúsculas y minúsculas (las minúsculas permanecerán en minúsculas) y todos los caracteres no alfabéticos deben permanecer intactos.

- Le pida al usuario una línea de texto para encriptar.
- Le pida al usuario un valor de cambio (un número entero del rango 1 al 25, nota: debes obligar al usuario a ingresar un valor de cambio válido (¡no te rindas y no dejes que los datos incorrectos te engañen!)).
- Imprime el texto codificado.

Prueba tu código utilizando los datos que te proporcionamos.

DATOS DE PRUEBA

Entrada Muestra:

```
abcxyzABCxyz 123
```

```
2
```

Salida Muestra:

```
cdezabCDEzab 123
```

Entrada Muestra:

```
The die is cast
```

```
25
```

Salida Muestra:

```
Sgd chd hr bzrs
```

```
# Ingresa el texto a encriptar
texto= input("Ingresa un mensaje: ")
# Ingresa un valor de cambio válido (repitelo hasta que tengas éxito)
shift = 0
while shift == 0:
    try:
        shift = int(input("Ingresa el valor de cambio del cifrado (1..25):
    "))
        if shift not in range(1,26):
            raise ValueError
    except ValueError:
        shift = 0
    if shift == 0:
        print("Valor de cambio inválido!")
cifrado = ''
for char in texto:
    # ¿Es un letra?
```

```
if char.isalpha():
    # cambia su código
    code = ord(char) + shift
    # encuentra el código de la primera letra (mayúscula o minúscula)
    if char.isupper():
        first = ord('A')
    else:
        first = ord('a')
    # hacer corrección
    code -= first
    code %= 26
    # agrega caracter codificado al mensaje
    cifrado += chr(first + code)
else:
    # agregar caracter original al mensaje
    cifrado += char

print(cifrado)
```