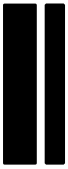# PRACTICAL 1

Program 1: Write a python program to demonstrate Caesar cipher.

Code:

```python
def int_to_char(x2):
    x1 = []
    for i in range(len(x2)):
        if (x2[i] == 0):
            x1.append('a')
        elif(x2[i] == 1):
            x1.append('b')
        elif(x2[i] == 2):
            x1.append('c')
        elif(x2[i] == 3):
            x1.append('d')
        elif(x2[i] == 4):
            x1.append('e')
        elif(x2[i] == 5):
            x1.append('f')
        elif(x2[i] == 6):
            x1.append('g')
        elif(x2[i] == 7):
            x1.append('h')
        elif(x2[i] == 8):
            x1.append('i')
        elif(x2[i] == 9):
            x1.append('j')
        elif(x2[i] == 10):
            x1.append('k')
```

```python
        elif(x2[i] == 11):
            x1.append('l')
        elif(x2[i] == 12):
            x1.append('m')
        elif(x2[i] == 13):
            x1.append('n')
        elif(x2[i] == 14):
            x1.append('o')
        elif(x2[i] == 15):
            x1.append('p')
        elif(x2[i] == 16):
            x1.append('q')
        elif(x2[i] == 17):
            x1.append('r')
        elif(x2[i] == 18):
            x1.append('s')
        elif(x2[i] == 19):
            x1.append('t')
        elif(x2[i] == 20):
            x1.append('u')
        elif(x2[i] == 21):
            x1.append('v')
        elif(x2[i] == 22):
            x1.append('w')
        elif(x2[i] == 23):
            x1.append('x')
        elif(x2[i] == 24):
            x1.append('y')
        elif(x2[i] == 25):
```
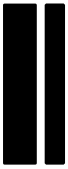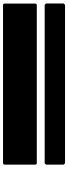
```python
        x1.append('z')


    str2 = ""
    a2 = 0
    for i in x1:
        str2 += i
        a2 += 1
        if a2 < len(x1):
            str2 += ""
        else:
            a = 1
    return str2
str1 = input("enter string: ")
key = int(input("enter key: "))
l1 = list(str1)
l2 = []
for i in range(len(l1)):
    if (l1[i] == 'a' or l1[i] == 'A'):
        l2.append(0)
    elif(l1[i] == 'b' or l1[i] == 'B'):
        l2.append(1)
    elif(l1[i] == 'c' or l1[i] == 'C'):
        l2.append(2)
    elif(l1[i] == 'd' or l1[i] == 'D'):
        l2.append(3)
    elif(l1[i] == 'e' or l1[i] == 'E'):
        l2.append(4)
    elif(l1[i] == 'f' or l1[i] == 'F'):
        l2.append(5)
```
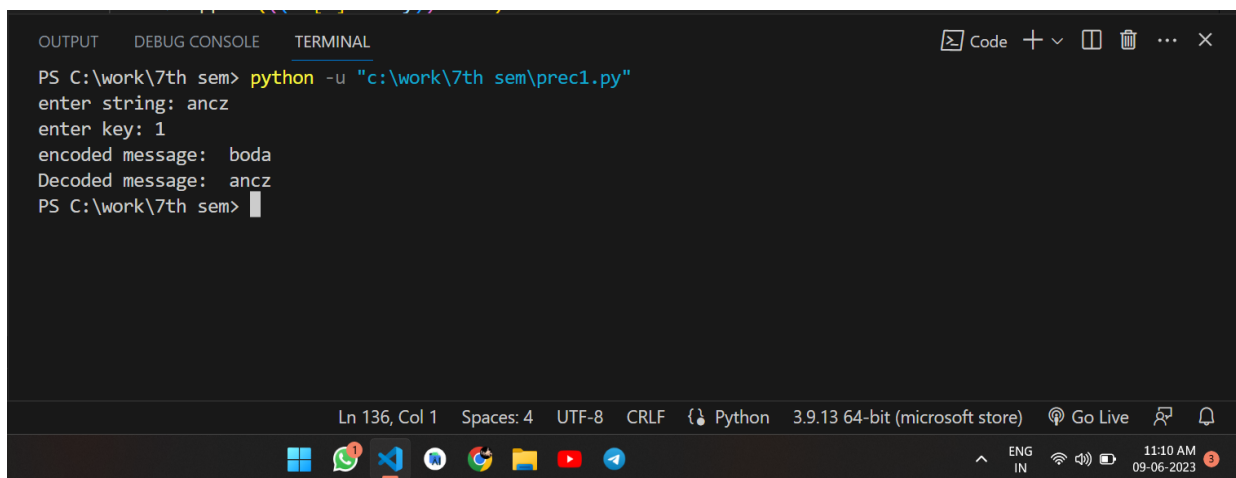
```
elif(l1[i] == 'g' or l1[i] == 'G'):

    l2.append(6)

elif(l1[i] == 'h' or l1[i] == 'H'):

    l2.append(7)

elif(l1[i] == 'i' or l1[i] == 'I'):

    l2.append(8)

elif(l1[i] == 'j' or l1[i] == 'J'):

    l2.append(9)

elif(l1[i] == 'k' or l1[i] == 'K'):

    l2.append(10)

elif(l1[i] == 'l' or l1[i] == 'L'):

    l2.append(11)

elif(l1[i] == 'm' or l1[i] == 'M'):

    l2.append(12)

elif(l1[i] == 'n' or l1[i] == 'N'):

    l2.append(13)

elif(l1[i] == 'o' or l1[i] == 'O'):

    l2.append(14)

elif(l1[i] == 'p' or l1[i] == 'P'):

    l2.append(15)

elif(l1[i] == 'q' or l1[i] == 'Q'):

    l2.append(16)

elif(l1[i] == 'r' or l1[i] == 'R'):

    l2.append(17)

elif(l1[i] == 's' or l1[i] == 'S'):

    l2.append(18)

elif(l1[i] == 't' or l1[i] == 'T'):

    l2.append(19)

elif(l1[i] == 'u' or l1[i] == 'U'):
```

```
        l2.append(20)
    elif(l1[i] == 'v' or l1[i] == 'V'):
        l2.append(21)
    elif(l1[i] == 'w' or l1[i] == 'W'):
        l2.append(22)
    elif(l1[i] == 'x' or l1[i] == 'X'):
        l2.append(23)
    elif(l1[i] == 'y' or l1[i] == 'Y'):
        l2.append(24)
    elif(l1[i] == 'z' or l1[i] == 'Z'):
        l2.append(25)
l3 = []
for i in range(len(l2)):
    l3.append(((l2[i] + key)) % 26)
l5 = []
for i in range(len(l3)):
    l5.append(((l3[i] - key)) % 26)
print("encoded message: ", int_to_char(l3))
print("Decoded message: ", int_to_char(l5))
```

output:

```
OUTPUT   DEBUG CONSOLE   TERMINAL                                    Code  + ∨  □  🗑  ⋯  ✕
PS C:\work\7th sem> python -u "c:\work\7th sem\prec1.py"
enter string: ancz
enter key: 1
encoded message:  boda
Decoded message:  ancz
PS C:\work\7th sem>
```