

Vigenere Cipher

Method 1

```

1 def encrypt(M, K):
2     E=[]
3     for i in range (len(M)):
4         E.append(alphabet[int(M[i])][int(K[i%len(key)])])
5     print (E)
6
7 def decrypt(M, K):
8     D = []
9     for i in range(len(M)):
10        col_index = alphabet[0].index(alphabet[int(K[i % len(K)])][0])
11        row_index = alphabet[col_index].index(alphabet[0][M[i]])
12        D.append(alphabet[0][row_index])
13    print(D)
14
15 alphabet = [['A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'],
16             ['B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A'],
17             ['C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B'],
18             ['D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C'],
19             ['E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D'],
20             ['F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E'],
21             ['G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F'],
22             ['H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G'],
23             ['I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H'],
24             ['J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I'],
25             ['K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J'],
26             ['L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K'],
27             ['M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L'],
28             ['N','O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M'],
29             ['O','P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N'],
30             ['P','Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O'],
31             ['Q','R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P'],
32             ['R','S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q'],
33             ['S','T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R'],
34             ['T','U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S'],
35             ['U','V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T'],
36             ['V','W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U'],
37             ['W','X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V'],
38             ['X','Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W'],
39             ['Y','Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X'],
40             ['Z','A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y']]
41
42
43 message = str(input("Input message in uppercase : "))
44 key = str(input("Input Key in uppercase : "))
45 temp = []
46 M=[]
47 K=[]
48
49 for i in range(len(message)):
50     if message[i].isalpha():
51         temp.append(ord(message[i]))
52     M.append(temp[-1] - ord('A'))
53
54 for i in range(len(key)):
55     if key[i].isalpha():
56         temp.append(ord(key[i]))
57     K.append(temp[-1] - ord('A'))
58
59 choice = int(input("1. Encrypt\n2. Decrypt\n"))
60
61 if (choice == 1):
62     encrypt(M, K)
63 if (choice == 2):
64     decrypt(M, K)

```

Input message in uppercase : GIVE MONEY

Input Key in uppercase : LOCK

1. Encrypt

2. Decrypt

1

['R', 'W', 'X', 'O', 'X', 'C', 'P', 'O', 'J']

Input message in uppercase : RWXO XCPOJ

Input Key in uppercase : LOCK

1. Encrypt

2. Decrypt

2

['G', 'I', 'V', 'E', 'M', 'O', 'N', 'E', 'Y']

Method 2

```
1 alphabet = ['A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z']
2
3 def encrypt(M, K):
4     E=[]
5     for i in range (len(M)):
6         E.append(alphabet[(int(M[i]) + int(K[i%len(key)])) % 26 ])
7     print (E)
8
9 def decrypt(M, K):
10    D = []
11    for i in range (len(M)):
12        D.append(alphabet[(int(M[i]) - int(K[i%len(key)]) + 26) % 26 ])
13    print(D)
14
15 message = str(input("Input message in uppercase : "))
16 key = str(input("Input Key in uppercase : "))
17 temp = []
18 M=[]
19 K=[]
20
21 for i in range(len(message)):
22     if message[i].isalpha():
23         temp.append(ord(message[i]))
24         M.append(temp[-1] - ord('A'))
25
26 for i in range(len(key)):
27     if key[i].isalpha():
28         temp.append(ord(key[i]))
29         K.append(temp[-1] - ord('A'))
30
31 choice = int(input("1. Encrypt\n2. Decrypt\n"))
32
33 if (choice == 1):
34     encrypt(M, K)
35 if (choice == 2):
36     decrypt(M, K)
```

```
Input message in uppercase : SHE IS LISTENING
Input Key in uppercase : PASCAL
1. Encrypt
2. Decrypt
1
['H', 'H', 'W', 'K', 'S', 'W', 'X', 'S', 'L', 'G', 'N', 'T', 'C', 'G']
```

```
Input message in uppercase : HHW KS WXSLGNTCG
Input Key in uppercase : PASCAL
1. Encrypt
2. Decrypt
2
['S', 'H', 'E', 'I', 'S', 'L', 'I', 'S', 'T', 'E', 'N', 'I', 'N', 'G']
```