In [1]:

```
def encrypt(plaintext,key):
    encryption_str = ''
    for i in plaintext:
        if i.isupper():
            temp = 65 + ((ord(i) - 65 + key) % 26)
            encryption_str = encryption_str + chr(temp)
        elif i.islower():
            temp = 97 + ((ord(i) - 97 + key) % 26)
            encryption_str = encryption_str + chr(temp)
        else:
            encryption_str = encryption_str + i
    return encryption_str

def decrypt(ciphertext,key):
    decryption_str = ''
    for i in ciphertext:
        if i.isupper():
            if ((ord(i) - 65 - key) < 0):
                temp = 65 + ((ord(i) - 65 - key + 26) % 26)
            else:
                temp = 65 + ((ord(i) - 65 - key) % 26)
            decryption_str = decryption_str + chr(temp)
        elif i.islower():
            if ((ord(i) - 97 - key) < 0):
                temp = 97 + ((ord(i) - 97 - key + 26) % 26)
            else:
                temp = 97 + ((ord(i) - 97 - key) % 26)
            decryption_str = decryption_str + chr(temp)
        else:
            decryption_str = decryption_str + i
    return decryption_str
```

In [2]:

```
def encryptRailFence(text, key):
    rail = [['\n' for i in range(len(text))]
    for j in range(key)]
    dir_down = False
    row, col = 0, 0
    for i in range(len(text)):
        if (row == 0) or (row == key - 1):
            dir_down = not dir_down
        rail[row][col] = text[i]
        col += 1
        if dir_down:
            row += 1
        else:
            row -= 1
    result = []
    for i in range(key):
        for j in range(len(text)):
            if rail[i][j] != '\n':
                result.append(rail[i][j])
    return("" . join(result))

def decryptRailFence(cipher, key):
    rail = [['\n' for i in range(len(cipher))] for j in range(key)]
    dir_down = None
    row, col = 0, 0
    for i in range(len(cipher)):
        if row == 0:
            dir_down = True
        if row == key - 1:
            dir_down = False
        rail[row][col] = '*'
        col += 1
        if dir_down:
            row += 1
        else:
            row -= 1
    index = 0
    for i in range(key):
        for j in range(len(cipher)):
            if ((rail[i][j] == '*') and (index < len(cipher))):
                rail[i][j] = cipher[index]
                index += 1
    result = []
    row, col = 0, 0
    for i in range(len(cipher)):
        if row == 0:
            dir_down = True
        if row == key-1:
            dir_down = False
        if (rail[row][col] != '*'):
            result.append(rail[row][col])
            col += 1
        if dir_down:
            row += 1
        else:
            row -= 1
    return("".join(result))
```

In [5]:

```
word=input("Enter the text to be ciphered: ")
key=int(input("Enter the key:"))
encrypted = encrypt(word,key)
print(encrypted)
encryptRail=encryptRailFence(encrypted,key)
print(encryptRail)
print("The text deciphered:")
decryptRail=decryptRailFence(encryptRail,key)
print(decryptRail)
decrypted=decrypt(decryptRail,key)
print(decrypted)
```

```
Enter the text to be ciphered: Warren
Enter the key:3
Zduuhq
Zhduqu
The text deciphered:
Zduuhq
Warren
```

In [ ]:

```

```