4. Write a python program for polyalphabetic substitution cipher uses a separate monoalphabetic substitution cipher for each successive letter of plaintext, depending on a key.

Code:

```
def vigenere_encrypt(plaintext, key):
    plaintext = plaintext.upper().replace(" ", "")
    key = key.upper()
    ciphertext = ""
    key_index = 0
    for ch in plaintext:
        shift = ord(key[key_index]) - ord('A')
        encrypted = chr((ord(ch) - ord('A') + shift) % 26 + ord('A'))
        ciphertext += encrypted
        key_index = (key_index + 1) % len(key)
    return ciphertext
def vigenere_decrypt(ciphertext, key):
    ciphertext = ciphertext.upper()
    key = key.upper()
    plaintext = ""
    key_index = 0
    for ch in ciphertext:
        shift = ord(key[key_index]) - ord('A')
        decrypted = chr((ord(ch) - ord('A') - shift) % 26 + ord('A'))
        plaintext += decrypted
        key_index = (key_index + 1) % len(key)
    return plaintext
```

```
# Example usage

text = "HELLO WORLD"

key = "KEY"

cipher = vigenere_encrypt(text, key)

plain = vigenere_decrypt(cipher, key)

print("Plaintext :", text)

print("Key       :", key)

print("Encrypted :", cipher)

print("Decrypted :", plain)
```