DATE: 25-08-2022

LAB 01: Working with classical ciphers

NAME	NAMAN CHOUDHARY
SRN	PES2UG20CS209
SECTION	D

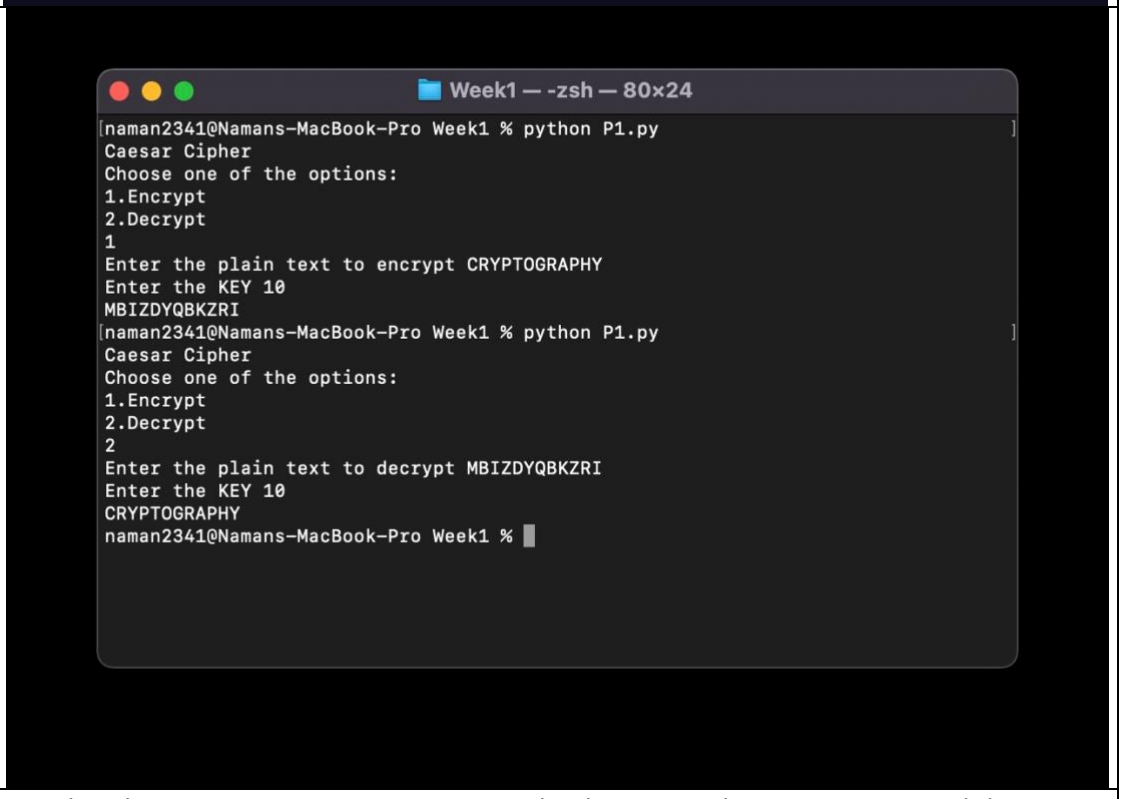
For the given questions, write a python code and attach the snapshots.

1.	For the given input, perform Caesar cipher encryption and decryption. Plain text: “CRYPTOGRAPHY” Key: 10
SOL	

```
#Caesar Cipher
#Example Plain Text: CRYPTOGRAPHY
#Key:10


def encrypt():
    plain=input("Enter the plain text to encrypt ")
    key=int(input("Enter the KEY "))
    lst=[]
    for i in plain:
        i=ord(i)-65
        i=(i+key)%26
        i=i+65
        lst.append(chr(i))
    str=''
    return(str.join(lst))


def decrypt():
    cipher=input("Enter the plain text to decrypt ")
    key=int(input("Enter the KEY "))
    lst=[]
    for i in cipher:
        i=ord(i)-65
        i=(i-key)%26
```

```
        i=i+65
        lst.append(chr(i))
    str=''
    return(str.join(lst))

print("Caesar Cipher")
op=int(input("Choose one of the options:\n1.Encrypt\n2.Decrypt\n"))
if op==1:
    print(encrypt())
else:
    print(decrypt())
```

SS



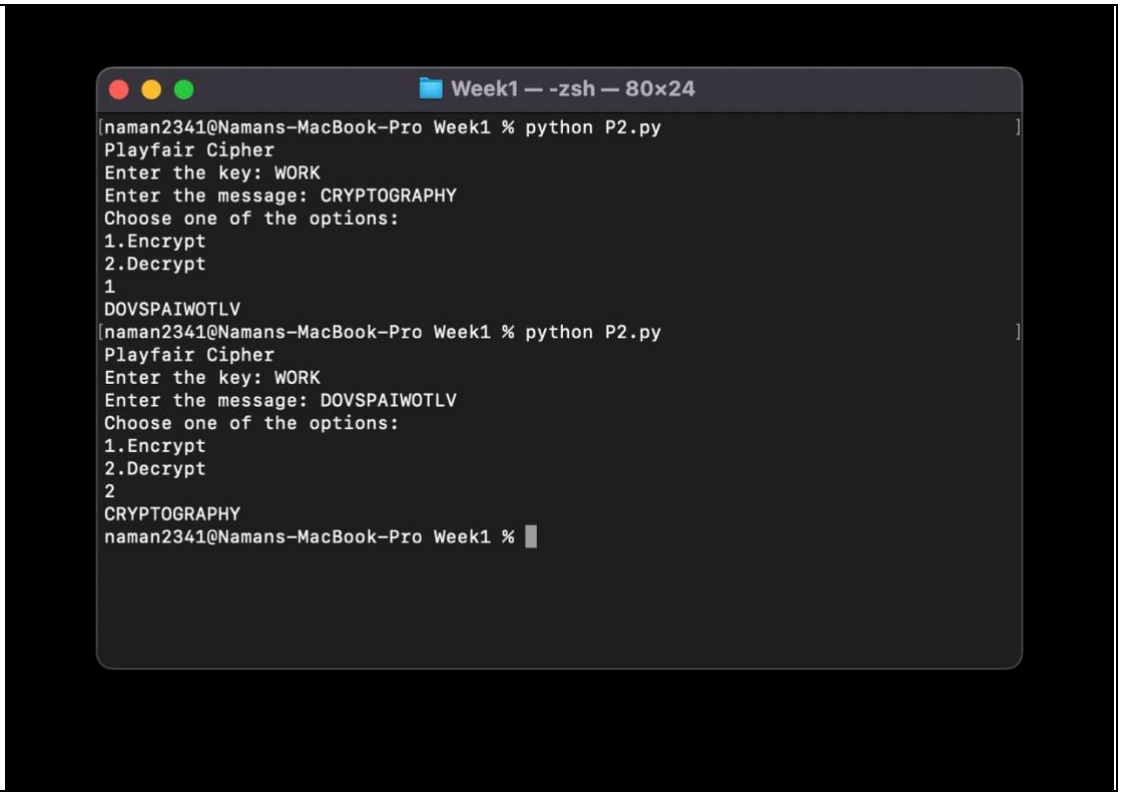
2. For the plaintext given in question 1, apply Play Fair cipher encryption with key “WORK”.

SOL

```
#Playfair Cipher
#Example Plain Text: <any>
#Key:WORK

def create_matrix(key):
    key = key.upper()
    matrix = [[0 for i in range(5)] for j in range(5)]
    letters_added = []
    row = 0
    col = 0

    for letter in key:
        if letter not in letters_added:
            matrix[row][col] = letter
            letters_added.append(letter)
        else:
            continue
        if (col == 4):
            col = 0
            row += 1
        else:
            col += 1

    for letter in range(65, 91):
        if letter == 74:
            continue
        if chr(letter) not in letters_added:
            letters_added.append(chr(letter))

    index = 0
    for i in range(5):
        for j in range(5):
            matrix[i][j] = letters_added[index]
            index += 1
```

```
    return matrix

def separate_same_letters(message):
    index = 0
    while (index < len(message)):
        l1 = message[index]
        if index == len(message)-1:
            message = message + 'X'
            index += 2
            continue
        l2 = message[index+1]
        if l1 == l2:
            message = message[:index+1] + "X" + message[index+1:]
        index += 2
    return message

def indexOf(letter, matrix):
    for i in range(5):
        try:
            index = matrix[i].index(letter)
            return (i, index)
        except:
            continue

def playfair(key, message, encrypt=True):
    inc = 1
    if encrypt == False:
        inc = -1
    matrix = create_matrix(key)
    message = message.upper()
    message = message.replace(' ', '')
    message = separate_same_letters(message)
    cipher_text = ''
    for (l1, l2) in zip(message[0::2], message[1::2]):
```

```
        row1, col1 = indexOf(l1, matrix)
        row2, col2 = indexOf(l2, matrix)
        if row1 == row2:
            cipher_text += matrix[row1][(col1+inc) %
                                       5] + matrix[row2][(col2+inc) % 5]
        elif col1 == col2:
            cipher_text += matrix[(row1+inc) % 5][col1] + \
                matrix[(row2+inc) % 5][col2]
        else:
            cipher_text += matrix[row1][col2] + matrix[row2][col1]

    return cipher_text

print("Playfair Cipher")
key = input("Enter the key: ")
message = input("Enter the message: ")
op=int(input("Choose one of the options:\n1.Encrypt\n2.Decrypt\n"))
if op==1:
    print(playfair(key, message, True))
else:
    print(playfair(key, message, False))
```

SS	

Week1 — -zsh — 80×24
naman2341@Namans-MacBook-Pro Week1 % python P2.py
Playfair Cipher
Enter the key: WORK
Enter the message: CRYPTOGRAPHY
Choose one of the options:
1.Encrypt
2.Decrypt
1
DOVSPAIWOTLV
naman2341@Namans-MacBook-Pro Week1 % python P2.py
Playfair Cipher
Enter the key: WORK
Enter the message: DOVSPAIWOTLV
Choose one of the options:
1.Encrypt
2.Decrypt
2
CRYPTOGRAPHY
naman2341@Namans-MacBook-Pro Week1 %