INFORMATION SECURITY

LAB MIDTERM

Name:

M Abdullah Azam

Reg no:

Sp24-Bse-034

Submitted to:

Mam Ambreen Gul

Date

21-10-2025

Question No # 05:

Conceptual Des And Aes

Part (a):

Write one similarity between DES and AES.

Answer:

**Both DES and AES are symmetric key block ciphers.
They use the same key for both encryption and decryption.
Both operate on fixed-size data blocks to secure information.**

Part(b):

What does CBC mode stand for in block ciphers?

Answer:

**CBC stands for Cipher Block Chaining mode.
In CBC, each plaintext block is XORed with the previous ciphertext block.
This makes encryption more secure by adding dependency between blocks.**

Part (c):

Why is AES faster than DES?

Answer:

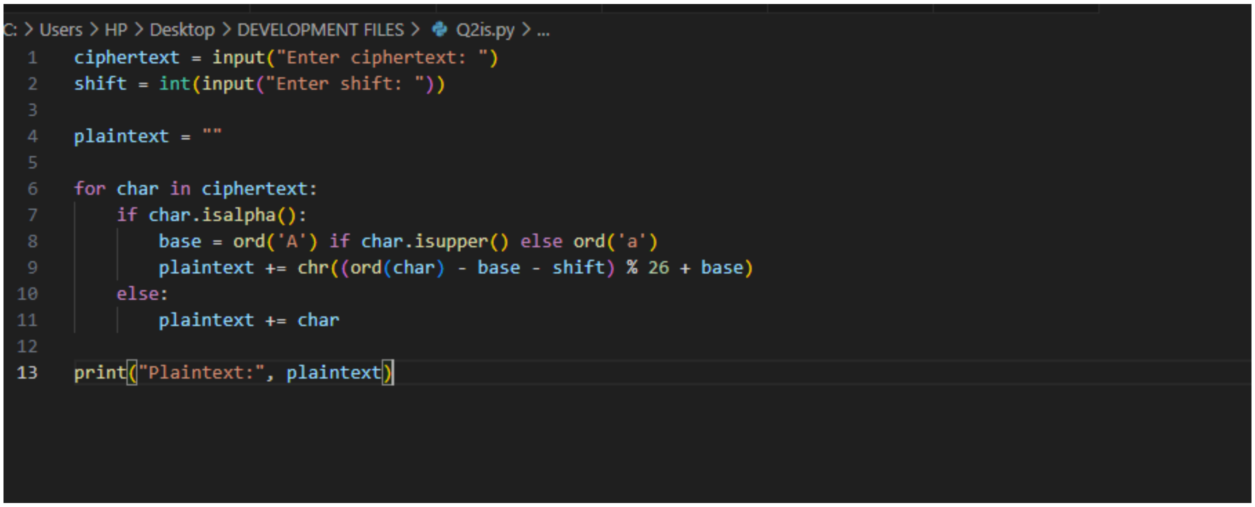
**AES is faster because it uses fewer, simpler rounds than DES.
Its algorithm is optimized for modern computer hardware.
AES also handles larger block sizes efficiently, improving speed.**

Question No # 02:

Write a Python program to decrypt a message that was encrypted using the Caesar Cipher. The program should take ciphertext (LXFOPVEFRNHR) and key (5) as input and display the plaintext.

Example:
Enter ciphertext: khoor
Enter shift: 3
Plaintext: hello

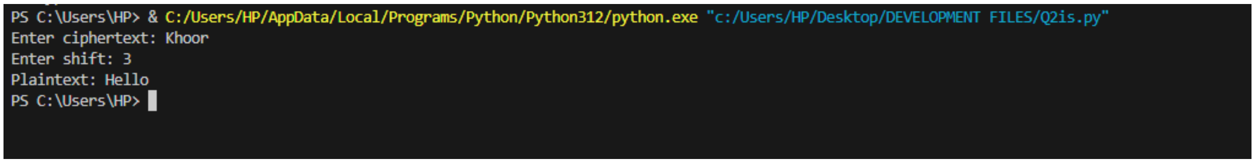
Answer:



C: > Users > HP > Desktop > DEVELOPMENT FILES > Q2is.py > ...

```python
ciphertext = input("Enter ciphertext: ")
shift = int(input("Enter shift: "))

plaintext = ""

for char in ciphertext:
    if char.isalpha():
        base = ord('A') if char.isupper() else ord('a')
        plaintext += chr((ord(char) - base - shift) % 26 + base)
    else:
        plaintext += char

print("Plaintext:", plaintext)
```

Output:



```
PS C:\Users\HP> & C:/Users/HP/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/HP/Desktop/DEVELOPMENT FILES/Q2is.py"
Enter ciphertext: Khoor
Enter shift: 3
Plaintext: Hello
PS C:\Users\HP> 
```

Question NO #03:

Write a Python program to decrypt a ciphertext using the Vigenère Cipher. Ask the user for ciphertext and key, and display the decrypted plaintext.

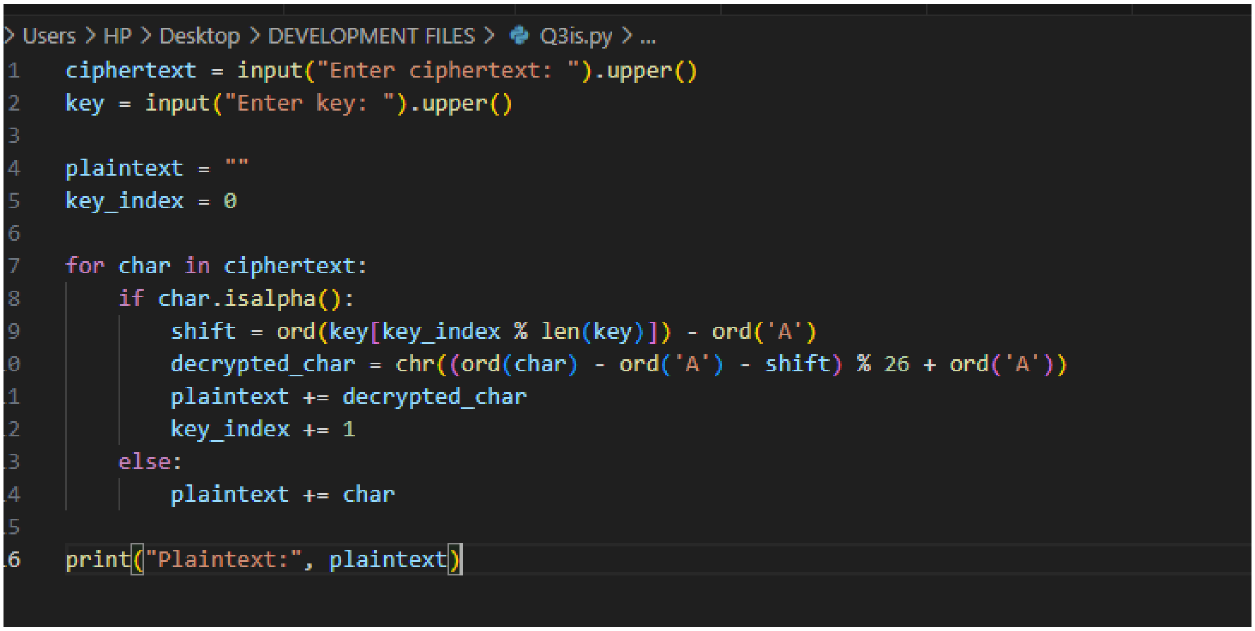
Example:

Enter ciphertext: LXFOPVEFRNHR

Enter key: LEMON

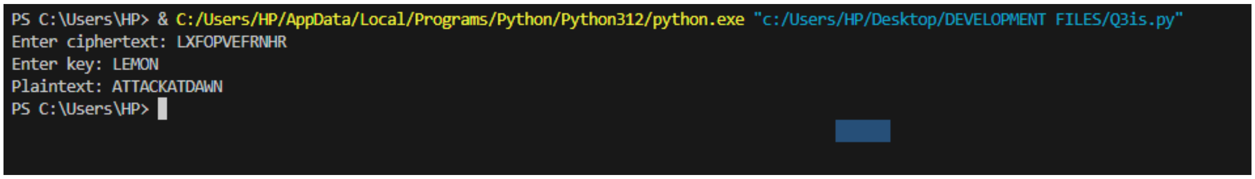
Plaintext: ATTACKATDAWN

Answer:



```python
ciphertext = input("Enter ciphertext: ").upper()
key = input("Enter key: ").upper()

plaintext = ""
key_index = 0

for char in ciphertext:
    if char.isalpha():
        shift = ord(key[key_index % len(key)]) - ord('A')
        decrypted_char = chr((ord(char) - ord('A') - shift) % 26 + ord('A'))
        plaintext += decrypted_char
        key_index += 1
    else:
        plaintext += char

print("Plaintext:", plaintext)
```

Output:



```
PS C:\Users\HP> & C:/Users/HP/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/HP/Desktop/DEVELOPMENT FILES/Q3is.py"
Enter ciphertext: LXFOPVEFRNHR
Enter key: LEMON
Plaintext: ATTACKATDAWN
PS C:\Users\HP> 
```

Question No # 04:

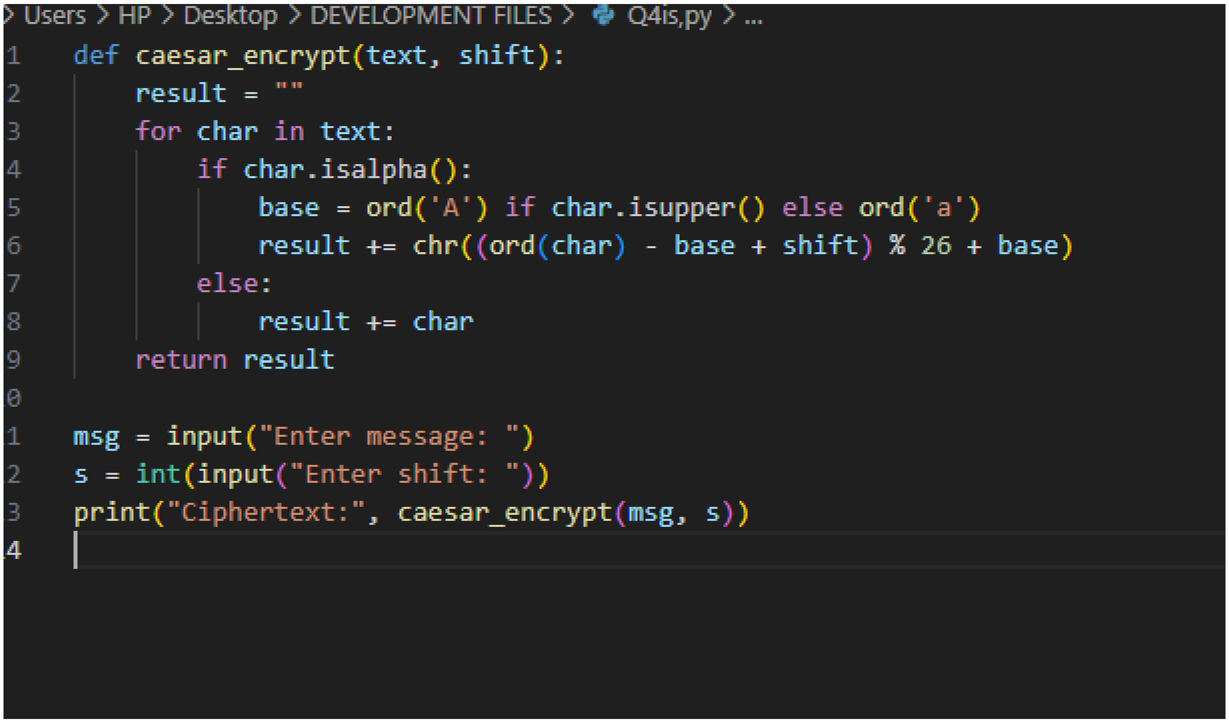
Answer:

Code Error:

result += chr(ord(char) + shift)

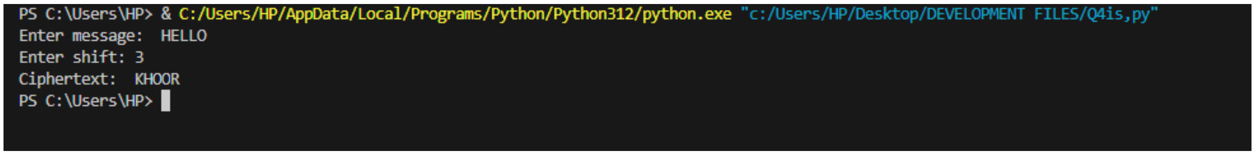
This line doesn’t wrap alphabets properly.

Fixed Code:



```python
def caesar_encrypt(text, shift):
    result = ""
    for char in text:
        if char.isalpha():
            base = ord('A') if char.isupper() else ord('a')
            result += chr((ord(char) - base + shift) % 26 + base)
        else:
            result += char
    return result

msg = input("Enter message: ")
s = int(input("Enter shift: "))
print("Ciphertext:", caesar_encrypt(msg, s))

```

Output:



```
PS C:\Users\HP> & C:/Users/HP/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/HP/Desktop/DEVELOPMENT FILES/Q4is,py"
Enter message:  HELLO
Enter shift: 3
Ciphertext:  KHOOR
PS C:\Users\HP> 
```

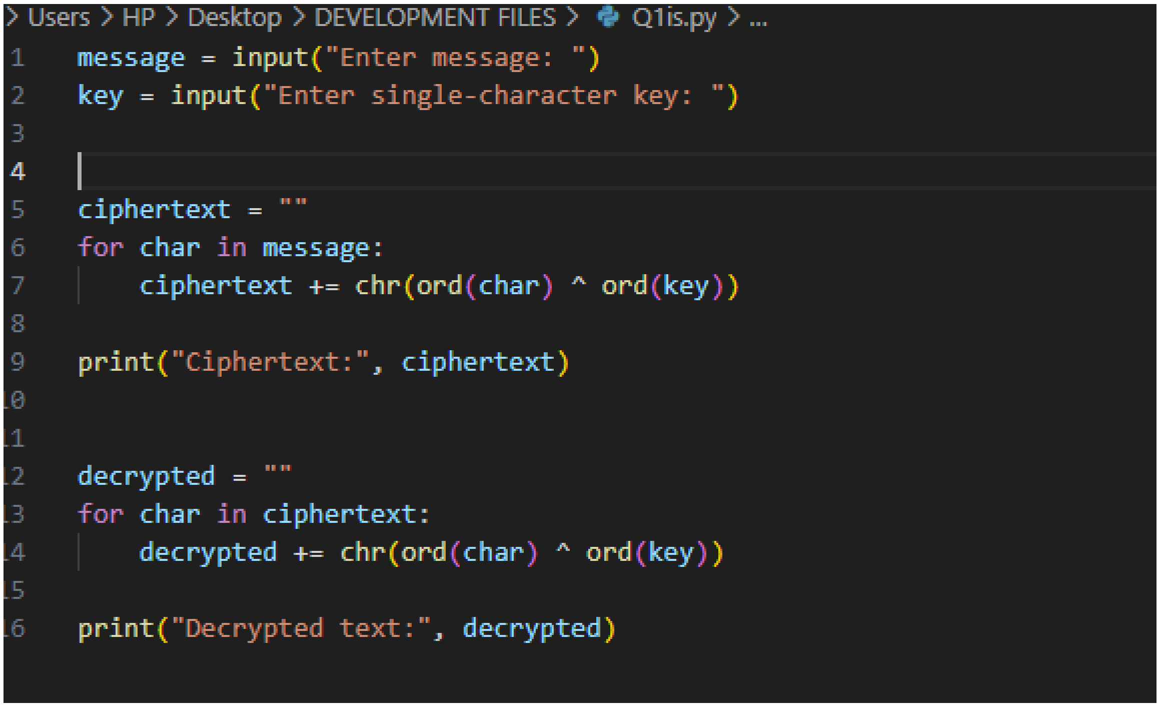
Question NO #01:

Write a Python program that performs both encryption and decryption using XOR operation.

Requirements:

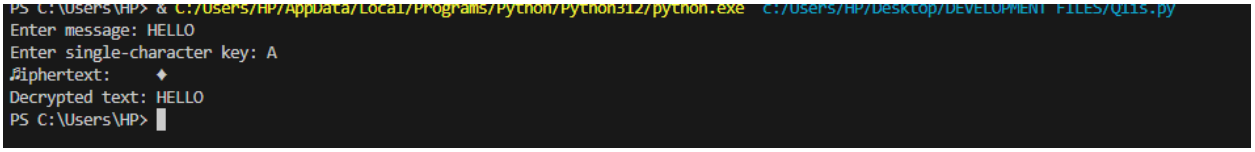
1. Ask the user for message and a single-character key.
2. Encrypt the message using XOR (ord() and chr()).
3. Decrypt it by applying XOR again with the same key.
4. Show both ciphertext and decrypted plaintext.

Answer:



```python
message = input("Enter message: ")
key = input("Enter single-character key: ")


ciphertext = ""
for char in message:
    ciphertext += chr(ord(char) ^ ord(key))

print("Ciphertext:", ciphertext)


decrypted = ""
for char in ciphertext:
    decrypted += chr(ord(char) ^ ord(key))

print("Decrypted text:", decrypted)
```

Output:



```
PS C:\Users\HP> & C:/Users/HP/AppData/Local/Programs/Python/Python312/python.exe  c:/Users/HP/Desktop/DEVELOPMENT FILES/Q1is.py
Enter message: HELLO
Enter single-character key: A
♬iphertext:     ♦
Decrypted text: HELLO
PS C:\Users\HP> █
```
