

Name - Siddharth Rawat Course - BCA SEM - 6th 'C'
University Roll No - 1121143 Subject - PBC-601-A CYBER LAWS
Date - 15/06/2021

4) WAP To Implement OTP (One Time Password)

```
import math, random
```

```
def generateOTP():
```

```
    digits = "0123456789"
```

```
    OTP = ""
```

```
    for i in range(4):
```

```
        OTP += digits[math.floor(random.random()*10)]
```

```
    return OTP
```

```
if __name__ == "__main__":
```

```
    print("OTP of 4 digits:", generateOTP())
```

Sid R

Name - Siddharth Rawat Course - BCA SEM-6th Student ID - 1821143
University Roll No - 1121143 Subject - PBC-601 - Information
Security & CYBER LAWS Date - 15/06/2021

Q3. WAP for the Encryption & decryption of Vignere

Cipher on the input plain text = 'Cryptography'
with a Key = 'Monarchy'

```
⇒ def generateKey(String, Key)
```

```
    Key = list(Key)
```

```
    if len(String) == len(Key)
```

```
        return Key
```

```
    else:
```

```
        for i in range(len(String) - len(Key)):
```

```
            Key.append(Key[i % len(Key)])
```

```
        return " ".join(Key)
```

```
def Encryption (String, Key):
```

```
    Ciphertext = []
```

```
    for i in range(len(String)):
```

```
        x = (ord(String[i]) + ord(Key[i])) % 2
```

```
        x += ord('A')
```

```
    Ciphertext.append(chr(x))
```

```
    return " ".join(Ciphertext)
```

```
def decryption (Ciphertext, Key):
```

Siddh.


```

    Orig_text = []
    for i in range(len(cipher_text)):
        x = (ord(cipher_text[i]) - ord(key[i])
              + 26) % 26
        x += ord('A')
        Orig_text.append(chr(x))
    return ''.join(Orig_text)

if if __name__ == "__main__":
    String = "Cryptography"
    Key = generatekey(String, Key)
    Ciphertext = encryption(String, Key)
    print("Cipher text: ", Ciphertext)
    print("Original / Decrypted text: ",
          decryption(Ciphertext, Key))

```

Siddh

Name - Siddharth Rawat Course - BCA SEM-6th 'C'
University Roll No - 1121143 Subject - PBC-601 Information
Security & Cyber Laws Date - 15/06/2021

Q5. WAP to implement encryption and decryption using
Caesar Cipher on the input plaintext = "Attack from
North"

⇒ Encryption Using Caesar cipher

```
def encrypt (String):
```

```
    Cipher = ''
```

```
    for char in String:
```

```
        if char == ' ':
```

```
            Cipher = Cipher + char
```

```
        elif char.isupper():
```

```
            Cipher = Cipher + chr((ord(char) + 3 - 65) %  
                                   26 + 65)
```

```
        else
```

```
            Cipher = Cipher + chr((ord(char) + 3 - 97) % 26 + 97)
```

```
    return Cipher
```

```
text = "Attack from North"
```

Decryption Using Caesar Cipher

```
def decrypt (String):
```

```
    Plain = ''
```

Sid R.

for char in string:

if char == ' ':

plain = plain + char

elif char.isupper():

plain = plain + chr((ord(char) - 3 - 65) % 26 + 65)

else

plain = plain + chr((ord(char) - 3 - 97) % 26 + 97)

return plain

text = "Dwulf nia Qrowk"

print (" ~~cipher~~ String", text)

print ("After ^{encryption} ~~decryption~~: ", decrypt(text))

print ("cipher string is ", text1)

print ("after decryption ", decrypt(text1))

Sidh.