Sankalp Pokhriyal
BCA - 6th - B
1121126 (45)
PBC - 601

F. Name - Shyam Singh.

Sankalp

Ans →

```
# import library
import math, random.

# function to genrate OTP
def genrateOTP():.

    # Declare a digits variable
    # which stores all digits
    digits = '0123456789"
    OTP = " "

    # length of password can be changed
    # by changing value in range
    for i. in range (4):.
        OTP += digits [math.floor (random.random()*10)]

    return OTP

# Driver code
if __name__ == "__main__":
    print ("OTP of 4 digits:", generateOTP())
```

Saukalp Pokhriyal
BCAG-B
1121126 (45)
PBC - 601

F. Name - Shyam Singh

Date
Page

Ceaser Cipher

```
# A python progam to illustrate caesar cipher Technique
def encrypt (text, s):
    result = " "

# traverse text
for i in range (len (text)):
    char = text [i]

    if (char . isupper()):
    result += chr((ord(char) + S - 65) % 26 + 65).

    else:
    result += chr ((ord (char) + S - 97) % 26
                                        + 97)

return result
    text = " Attack from north".
    S = 4.
        print " text:" + text
        print "shift:" + str (s)
        print "(ipher" + encrypt (text, s).
```

Saukalp Pokhriyal
BCA - 6 - B
1121126 (45)
PBC - 601

F. Name → Shyam Singh.

classmate
Date
Page

Saukalp

Ans3.

Vigenere Cipher

```
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 originalText (cipher_text , Key):
    orig_text = []
    for i in range (len (cipher_text)):
        x = (ord (cipher_text [i])
            ord (Key [i] + 26) % 26
        x + = ord ('A').
        return ("" . join (orig_text)).

                                    ..)).

if __name__ == "__main__":
    String = " BCAB "
    Keyword = " Saukalp "
    Key = generateKey (String , Keyword)
    cipher_text = cipherText (String , Key).
print (" ciphertext : " , cipher_text)
print (" Original text : ", originalText (cipher_text,
                                    key)).
```