Name- Rahul Singh
Roll. no. 1121106 (25)
Section- B
Subject - Information security and Cyber laws

Que-1 The three security aspects of Google Accounts Are :-

1- Prevent unauthorized access to your Account : Advance protection requires security keys for sign in to help protect your google data, likes emails, documents contacts or other personal google data. Even if hacker has your username and password they can't sign in without your security key.

2. Provides extra protection from Harmful downloads - Advanced protection performs extra checks on the downloads. When downloading a file that may be harmful it. notifies you or blocks the download on your Android phones, only apps from verified stores are allowed.

3. Keep Your Personal Information Secure - To prevent ~~a~~ unauthorized access, Advanced protection only access google apps and verified third party apps to access your google account data. and only with your permission.

Rahul

Que-4 Write a program to implement OTP

```
# import library
import math, random

# function to generate OTP
def generateOTP() :

# 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 ())
```

Output:

OTP of 4 digits: 3211

Rahul

Que-Q5

```
def encrypt ()
 cipher = ''
String = 'Attack from North'
for char in string:
if char == '':
  cipher = cipher + char
elif char is upper ():
  cipher = cipher + chr ((ord (char) + 3 - 65) %
                                    26 + 65)
else:
  cipher = cipher + chr ((ord (char) + 3 - 97) %
                                    26 + 97)
return cipher
print ("original string", string)
print ("After encryption", encrypt ())
str = cipher
def decrypt ()
plain = ''
 for char in string:
if char = '';
  plain = plain + char
elif char is supper ():
plain = plain + chr ((ord (char) - 3 - 65) % 26 + 65)
else
  plain = plain + chr ((ord (char) - 3 - 97) % 26 + 97)
return plain
 print ("cipher string", str)
 print ("After decryption", decrypt)
```

Rahul