**Practical 1A**

**Aim: A simple client class that generates the private and public keys by using the built-in Python RSA algorithm and test it.**

import hashlib

import random

import binascii

import datetime

import collections

from Crypto.PublicKey import RSA

from Crypto import Random

from Crypto.Cipher import PKCS1\_v1\_5

class Client:

   def \_\_init\_\_(self):

      random = Random.new().read

      self.\_private\_key = RSA.generate(1024, random)

      self.\_public\_key = self.\_private\_key.publickey()

      self.\_signer = PKCS1\_v1\_5.new(self.\_private\_key)

   @property

   def identity(self):

      return binascii.hexlify(self.\_public\_key.exportKey(format='DER')).decode('ascii')

Veena = Client()

print ("sender ",Veena.identity)