AIM:

5.Write a program for simple RSA algorithm to encrypt and decrypt the data.

IMPLEMENTATION:

```
import java.io.*;
import java.math.*;
import java.util.*;
/*
 * Java program for RSA asymmetric cryptographic algorithm.
 * For demonstration, values are
 * relatively small compared to practical application
 */
public class GFG {
  public static double gcd(double a, double h)
  {
    /*
         * This function returns the gcd or greatest common
         * divisor
         */
    double temp;
    while (true) {
      temp = a % h;
      if (temp == 0)
        return h;
      a = h;
      h = temp;
    }
  }
  public static void main(String[] args)
  {
    double p = 3;
    double q = 7;

    // Stores the first part of public key:
    double n = p * q;

    // Finding the other part of public key.
    // double e stands for encrypt
    double e = 2;
    double phi = (p - 1) * (q - 1);
    while (e < phi) {
      /*
             * e must be co-prime to phi and
             * smaller than phi.
             */
      if (gcd(e, phi) == 1)
        break;
      else
        e++;
    }
    int k = 2; // A constant value
    double d = (1 + (k * phi)) / e;

    // Message to be encrypted
    double msg = 12;

    System.out.println("Message data = " + msg);

    // Encryption c = (msg ^ e) % n
    double c = Math.pow(msg, e);
    c = c % n;
    System.out.println("Encrypted data = " + c);

    // Decryption m = (c ^ d) % n
    double m = Math.pow(c, d);
    m = m % n;
    System.out.println("Original Message Sent = " + m);
  }
}

// This code is contributed by Pranay Arora
```