```
public class SimpleRSA {
    public static void main(String[] args) {
        // Step 1: Prime numbers (p and q)
        int p = 3, q = 7;

        // Step 2: Compute n and phi
        int n = p * q;
        int phi = (p - 1) * (q - 1);

        // Step 3: Choose e (Public Key Part)
        int e = 2;
        while (gcd(e, phi) != 1) {
            e++;
        }

        // Step 4: Compute d (Private Key Part)
        int k = 2; // constant multiplier
        int d = (1 + (k * phi)) / e;

        // Step 5: Message to encrypt
        int msg = 12;
        System.out.println("Message: " + msg);

        // Step 6: Encrypt the message
        int encrypted = (int) (Math.pow(msg, e) % n);
        System.out.println("Encrypted: " + encrypted);

        // Step 7: Decrypt the message
        int decrypted = (int) (Math.pow(encrypted, d) % n);
        System.out.println("Decrypted: " + decrypted);
    }

    // Function to calculate GCD
    public static int gcd(int a, int b) {
        while (b != 0) {
            int temp = b;
            b = a % b;
            a = temp;
        }
        return a;
    }
}
```