Question:

Write a Program in Java to input a number and check whether it is an Evil Number or not.

Evil Number : An Evil number is a positive whole number which has even number of 1’s in its binary equivalent.

Example: Binary equivalent of 9 is 1001, which contains even number of 1’s.  
A few evil numbers are 3, 5, 6, 9….

Design a program to accept a positive whole number and find the binary equivalent of the number and count the number of 1’s in it and display whether it is a Evil number or not with an appropriate message. Output the result in format given below:

Example 1  
INPUT : 15  
BINARY EQUIVALENT : 1111  
NO. OF 1’s : 4  
OUTPUT : EVIL NUMBER

Example 2  
INPUT : 26  
BINARY EQUIVALENT : 11010  
NO. OF 1’s : 3  
OUTPUT : NOT AN EVIL NUMBER

import java.util.\*;

class EvilNumber

{

    String toBinary(int n) // Function to convert a number to Binary

    {

        int r;

        String s=""; //variable for storing the result

        char dig[]={'0','1'}; //array storing the digits (as characters) in a binary number system

        while(n>0)

            {

                r=n%2; //finding remainder by dividing the number by 2

                s=dig[r]+s; //adding the remainder to the result and reversing at the same time

                n=n/2;

            }

        return s;

    }

    int countOne(String s) // Function to count no of 1's in binary number

    {

        int c = 0, l = s.length();

        char ch;

        for(int i=0; i<l; i++)

        {

            ch=s.charAt(i);

            if(ch=='1')

            {

                c++;

            }

        }

        return c;

    }

    public static void main(String args[])

    {

        EvilNumber ob = new EvilNumber();

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a positive number : ");

        int n = sc.nextInt();

        String bin = ob.toBinary(n);

        System.out.println("Binary Equivalent = "+bin);

        int x = ob.countOne(bin);

        System.out.println("Number of Ones = "+x);

        if(x%2==0)

            System.out.println(n+" is an Evil Number.");

        else

            System.out.println(n+" is Not an Evil Number.");

    }

}

#### **Question:**

Write a Program in Java to input a number and check whether it is a **Fascinating Number** or not..

**Fascinating Numbers :** Some numbers of 3 digits or more exhibit a very interesting property. The property is such that, when the number is multiplied by 2 and 3, and both these products are concatenated with the original number, all digits from 1 to 9 are present exactly once, regardless of the number of zeroes.

Let’s understand the concept of Fascinating Number through the following example:

Consider the number 192,  
192 x 1 = 192  
192 x 2 = 384  
192 x 3 = 576

Concatenating the results : **192384576**

It could be observed that ‘192384576’ consists of all digits from 1 to 9 exactly once. Hence, it could be concluded that 192 is a Fascinating Number.

Some examples of fascinating Numbers are : 192, 219, 273, 327, 1902, 1920, 2019 etc.

import java.util.\*;

class FascinatingNumber

{

    boolean isUnique(String q)

    {

        int A[] = {0,0,0,0,0,0,0,0,0,0}; //to store frequency of every digit from '0' to '9'

        int i, flag = 0;

        char ch;

        for(i=0; i<q.length(); i++)

        {

            ch = q.charAt(i);

            A[ch-48]++;

            /\*  increasing A[5] if ch='5' as '5'-48 = 53-48=5

             \*  (ASCII values of '0' to '9' are 48 to 57) \*/

        }

        for(i=1; i<10; i++)

        {

            //checking if every digit from '1' to '9' are present exactly once or not

            if(A[i]!=1)

            {

                flag = 1; //flag is set to 1 if frequency is not 1

                break;

            }

        }

        if(flag == 1)

            return false;

        else

            return true;

    }

    public static void main(String args[])

    {

        Scanner sc = new Scanner(System.in);

        FascinatingNumber ob = new FascinatingNumber();

        System.out.print("Enter a number : ");

        int n = sc.nextInt();

        String p = Integer.toString(n); //converting the number to String

        if(p.length()<3)

            System.out.println("Number should be of atleast 3 digits.");

        else

        {

            String s = Integer.toString(n\*1) + Integer.toString(n\*2) + Integer.toString(n\*3);

            /\*  Joining the first, second and third multiple of the number

             \*  by converting them to Strings and concatenating them\*/

            if(ob.isUnique(s))

                System.out.println(n+" is a Fascinating Number.");

            else

                System.out.println(n+" is not a Fascinating Number.");

        }

    }

}

#### **Question:**

Write a Program in Java to input a word and print its anagrams..

**Note:** Anagrams are words made up of all the characters present in the original word by re-arranging the characters.

**Example:** Anagrams of the word TOP are: TOP, TPO, OPT, OTP, PTO and POT

import java.util.\*;

class Anagrams

{

    int c = 0;

    void input()throws Exception

    {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a word : ");

        String s = sc.next();

        System.out.println("The Anagrams are : ");

        display("",s);

        System.out.println("Total Number of Anagrams = "+c);

    }

    void display(String s1, String s2)

    {

        if(s2.length()<=1)

        {

            c++;

            System.out.println(s1+s2);

        }

        else

        {

            for(int i=0; i<s2.length(); i++)

            {

                String x = s2.substring(i, i+1);

                String y = s2.substring(0, i);

                String z = s2.substring(i+1);

                display(s1+x, y+z);

            }

        }

    }

    public static void main(String args[])throws Exception

    {

        Anagrams ob=new Anagrams();

        ob.input();

    }

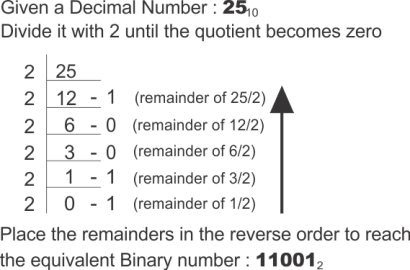
}

Question:

Write a Program in Java to input a number in Decimal number system and convert it into its equivalent number in the Binary number system.

Note: Binary Number system is a number system which can represent a number in any other number system in terms of 0 and 1 only. This number system consists of only two basic digits i.e. 0 and 1.

For Example: 25 in the Decimal number system can be represented as 11001 in the Binary number system.

[](http://i2.wp.com/www.guideforschool.com/wp-content/uploads/2013/08/decimal-to-binary-number-conversion.png)

import java.io.\*;

class Dec2Bin

{

    public static void main(String args[])throws IOException

    {

        BufferedReader br=new BufferedReader (new InputStreamReader(System.in));

        System.out.print("Enter a decimal number : ");

        int n=Integer.parseInt(br.readLine());

        int r;

        String s=""; //variable for storing the result

        char dig[]={'0','1'}; //array storing the digits (as characters) in a binary number system

        while(n>0)

            {

                r=n%2; //finding remainder by dividing the number by 2

                s=dig[r]+s; //adding the remainder to the result and reversing at the same time

                n=n/2;

            }

        System.out.println("Output = "+s);

    }

}