Write a program to convert numbers into words using Enumerations with constructors, methods and instance variables.(INPUT RANGE-0 TO 99999)

```java
import java.util.Scanner;

enum NumString {
    ZERO(0), ONE(1), TWO(2), THREE(3), FOUR(4), FIVE(5), SIX(6), SEVEN(7), EIGHT(8), NINE(9), TEN(10), ELEVEN(11),
    TWELVE(12), THIRTEEN(13), FOURTEEN(14), FIFTEEN(15), SIXTEEN(16), SEVENTEEN(17), EIGHTEEN(18), NINETEEN(19),
    TWENTY(20), THIRTY(30), FORTY(40), FIFTY(50), SIXTY(60), SEVENTY(70), EIGHTY(80), NINETY(90);

    int value;

    NumString(int value) {
        this.value = value;
    }

    static String getNumString(int n) {
        for (NumString v : values())
            if (v.value == n)
                return v.toString();
        return "";
    }

    static String convert(int n) {
        if (n <= 20)
            return getNumString(n);

        int rem = n % 10;
        if (rem == 0)
            return getNumString(n);
        else
            return getNumString((n / 10) * 10) + " " + getNumString(rem);
    }
}
```

```java
public class NumToWordNew {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number (0-99999): ");
        int num = sc.nextInt();
        if (num < 0 || num > 99999) {
            System.out.println("Number out of range");
            return;
        }
        if (num == 0) {
            System.out.println(NumString.ZERO.toString());
            return;
        }

        String word = "";

        if (num >= 1000) {
            word += NumString.convert(num / 1000) + " THOUSAND ";
            num %= 1000;
        }

        if (num >= 100) {
            word += NumString.convert(num / 100) + " HUNDRED ";
            num %= 100;
        }

        if (num > 0)
            word += NumString.convert(num);

        System.out.println("Number in words: " + word);
    }
}
```