**Logic**The program reads a number from the console and passes it on to the method numberToWord. This method declares two arrays : one holds the word representation of numbers from 0 to 19(one, two, three, four etc. ) and other which holds the tens representation of numbers(ten, twenty, thirty etc.).  
numberToWord is a recursive method which has multiple if-else conditions which divide the number by 10, 100, 1000, 10000 and so on. This can be increased depending on the maximum limit of the number.  
Based on the multiple of ten which divides the number, it calculates the quotient and calls itself again with the leftmost digit removed for converting the next digit to a word.

Suppose the number is 3435, the method will check that the number is divisible by 1000. Hence the if condition (number / 1000 > 0) is executed. It will calculate the quotient of division **3435 / 1000 = 3**.  
The if condition which divides the number by 1000 knows that it has divided the number by thousand, it appends the word “**thousand**” after the quotient and calls the method again with the remaining number 435.  
The remaining number is calculated by taking modulus of number by the multiple of 10 which divides the number(1000, in this case). Thus modulus is calculated as **3435 % 1000 = 435**.

**Let’s tweak in**

1. The program reads the number as an integer. Thus, it can handle input values of length 9 only which means till 100s million.  
   If you want to increase the range further, use bigger data types such as long, BigInteger etc.
2. The program converts the numbers into words based on International numbering system. A little modification is needed to convert it into Indian numbering system.
3. If minus symbol is not removed when the number is negative, the program will give a java.lang.StackOverflowError since the number will remain negative and it will keep on executing the same if condition which checks whether the number is less than 0 and the method will keep on calling itself in an infinite loop