Question: https://leetcode.com/problems/integer-to-english-words/

So since we arre dealing with international naming convention, we will get set of three digits at each stage to handle that is less than thousand, less than million and less than billion and above it.

Now we use a while loop to check that.

Now inside that set of three digits there will be hundred, tens, and ones.

In tens eleven twelve thirteen fourteen fifteen, sixteen, seventeen, eighteen, nineteen this are exceptions.

So we use a helper function to deal this set of 3 numbers.

Code:  
class Solution {

private final String[] LESS\_THAN\_20 = {"", "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine", "Ten", "Eleven", "Twelve", "Thirteen", "Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen"};

private final String[] TENS = {"", "Ten", "Twenty", "Thirty", "Forty", "Fifty", "Sixty", "Seventy", "Eighty", "Ninety"};

private final String[] THOUSANDS = {"", "Thousand", "Million", "Billion"};

public String numberToWords(int num) {

if (num == 0) return "Zero";

int i = 0;

String words = "";

while (num > 0) {

if (num % 1000 != 0)

words = helper(num % 1000) +THOUSANDS[i%4] + " " + words;

num /= 1000;

i++;

}

return words.trim();

}

private String helper(int num) {

if (num == 0)

return "";

else if (num < 20)

return LESS\_THAN\_20[num] + " ";

else if (num < 100)

return TENS[num / 10] + " " + helper(num % 10);

else

return LESS\_THAN\_20[num / 100] + " Hundred " + helper(num % 100);

}

}

Github Link :<https://lnkd.in/ecwtJeaz>