

Question : Leetcode for Integer to english words

Solution: `class Solution {`

```
    public String numberToWords(int num) {  
        return num == 0 ? "Zero" : helper(num);  
    }  
  
    private final String[] belowTwenty = {"", "One", "Two", "Three",  
"Four", "Five", "Six", "Seven", "Eight",  
"Nine", "Ten", "Eleven", "Twelve", "Thirteen",  
"Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen",  
"Nineteen"};  
    private final String[] tens = {"", "Twenty", "Thirty", "Forty",  
"Fifty", "Sixty", "Seventy", "Eighty", "Ninety"};  
  
    private String helper(int num) {  
        StringBuilder s = new StringBuilder();  
  
        if (num < 20)  
            s.append(belowTwenty[num]);  
        else if (num < 100)  
            s.append(tens[num / 10]).append(" ").append(belowTwenty[num % 10]);  
        else if (num < 1000)  
            s.append(helper(num / 100)).append(" Hundred ").append(helper(num % 100));  
        else if (num < 1000000)  
            s.append(helper(num / 1000)).append(" Thousand ").append(helper(num % 1000));  
        else if (num < 1000000000)
```

```

        s.append(helper(num / 1000000)).append(" Million ").append(helper(num % 1000000));
    else
        s.append(helper(num / 1000000000)).append(" Billion ").append(helper(num %
1000000000));

    return s.toString().trim();
}
}

```

Question: Leet code for Sum power of three

Solution:

```

class Solution {
    public boolean checkPowersOfThree(int n) {
        while (n > 1) {
            int r = n % 3;
            if (r == 2)
                return false;
            n /= 3;
        }

        return true;
    }
}

```