The **numeric value** of a **lowercase character** is defined as its position (1-indexed) in the alphabet, so the numeric value of a is 1, the numeric value of b is 2, the numeric value of c is 3, and so on.

The **numeric value** of a **string** consisting of lowercase characters is defined as the sum of its characters' numeric values. For example, the numeric value of the string "abe" is equal to 1 + 2 + 5 = 8.

You are given two integers n and k. Return *the****lexicographically smallest string****with****length****equal to n and****numeric value****equal to k.*

Note that a string x is lexicographically smaller than string y if x comes before y in dictionary order, that is, either x is a prefix of y, or if i is the first position such that x[i] != y[i], then x[i] comes before y[i] in alphabetic order.

**Example 1:**

**Input:** n = 3, k = 27

**Output:** "aay"

**Explanation:** The numeric value of the string is 1 + 1 + 25 = 27, and it is the smallest string with such a value and length equal to 3.

**Example 2:**

**Input:** n = 5, k = 73

**Output:** "aaszz"

**Constraints:**

* 1 <= n <= 105
* n <= k <= 26 \* n