

```

class Solution {
    public int minSubArrayLen(int s, int[] nums) {
        int l = 0, r = 0, sum = 0, res = Integer.MAX_VALUE;

        while (r < nums.length) {
            sum += nums[r++];
            while (sum >= s) {
                res = Math.min(res, r - l);
                sum -= nums[l++];
            }
        }

        return res == Integer.MAX_VALUE ? 0 : res;
    }
}

```

Given an array of **n** positive integers and a positive integer **s**, find the minimal length of a **contiguous** subarray of which the sum  $\geq$  **s**. If there isn't one, return 0 instead.

### Example:

**Input:** `s = 7, nums = [2,3,1,2,4,3]`

**Output:** `2`

**Explanation:** the subarray `[4,3]` has the minimal length under the problem constraint.