Given an integer array nums with possible **duplicates**, randomly output the index of a given target number. You can assume that the given target number must exist in the array.

Implement the Solution class:

* Solution(int[] nums) Initializes the object with the array nums.
* int pick(int target) Picks a random index i from nums where nums[i] == target. If there are multiple valid i's, then each index should have an equal probability of returning.

**Example 1:**

Input  
["Solution", "pick", "pick", "pick"]  
[[[1, 2, 3, 3, 3]], [3], [1], [3]]  
Output  
[null, 4, 0, 2]  
  
Explanation  
Solution solution = new Solution([1, 2, 3, 3, 3]);  
solution.pick(3); // It should return either index 2, 3, or 4 randomly. Each index should have equal probability of returning.  
solution.pick(1); // It should return 0. Since in the array only nums[0] is equal to 1.  
solution.pick(3); // It should return either index 2, 3, or 4 randomly. Each index should have equal probability of returning.

**Constraints:**

* 1 <= nums.length <= 2 \* 104
* -231 <= nums[i] <= 231 - 1
* target is an integer from nums.
* At most 104 calls will be made to pick.