**[Unbounded Knapsack](https://practice.geeksforgeeks.org/problems/knapsack-with-duplicate-items4201/1?utm_source=gfg&utm_medium=article&utm_campaign=bottom_sticky_on_article)**

Given a set of **N** items, each with a weight and a value, represented by the array **w[]** and **val[]** respectively. Also, a knapsack with weight limit **W**.  
The task is to fill the knapsack in such a way that we can get the maximum profit. Return the maximum profit.  
Note: Each item can be taken any number of times.

**Example 1:**

**Input:** N = 2, W = 3

val[] = {1, 1}

wt[] = {2, 1}

**Output:** 3

**Explanation:**

1.Pick the 2nd element thrice.

2.Total profit = 1 + 1 + 1 = 3. Also the total

 weight = 1 + 1 + 1 = 3 which is <= W.

**Example 2:**

**Input:** N = 4, W = 8

val[] = {1, 4, 5, 7}

wt[] = {1, 3, 4, 5}

**Output:** 11

**Explanation:** The optimal choice is to

pick the 2nd and 4th element.

**Your Task:**  
You do not need to read input or print anything. Your task is to complete the function **knapSack()** which takes the values N, W and the arrays val[] and wt[] as input parameters and returns the maximum possible value.

**Expected Time Complexity:** O(N\*W)  
**Expected Auxiliary Space:**O(W)

**Constraints:**  
1 ≤ N, W ≤ 1000  
1 ≤ val[i], wt[i] ≤ 100