Equal Sum partition

1)Check the sum of the array divisible by 2 or not

Public class Solution {

Boolean dp[][];

//delared it as Boolean instead of boolean because hear in wrapper class initial value will be null;

public boolean callPartition(int[] nums,int n,int sum)

{

if(n==0 && sum!=0)

return false;

if(sum==0)

return true;

if(dp[n][sum]!=null)

return dp[n][sum];

if(nums[n-1]<=sum)

return dp[n][sum]=callPartition(nums,n-1,sum)||callPartition(nums,n-1,sum-nums[n-1]);

else

return dp[n][sum]=callPartition(nums,n-1,sum);

}

public boolean canPartition(int[] nums) {

int i,sum=0;

for(i=0;i<nums.length;i++)

sum+=nums[i];

//check if the sum is even.. if its odd then there cant be two equal partition.

if(sum%2!=0)

return false;

sum=sum/2;

//but dividing the sum by 2 this becomes same as subset sum.

dp=new Boolean[nums.length+1][sum+1];

boolean res=callPartition(nums,nums.length,sum);

return res;

}

}