Question:https://leetcode.com/problems/palindromic-substrings/

I have used DP to solve the problem like the previous palindrome problem we are constructng a 2d matrix and checking if substring from i to j is palindrome or not, if palindrome then we fill the value with 1, else ) so now the number of ones in the matrix is the answer.

I used DP to solve it because we can solve multiple palindrome string problems using this approach.

Code:  
class Solution {

public int countSubstrings(String s) {

int dp[][] = new int[s.length()][s.length()];

int count=0;

for(int i=0; i<s.length(); i++){

int z=0, j=i;

while(z<s.length() && j<s.length()){

System.out.println(z+" "+j);

if(z==j){

dp[z][j]=1;

}

else if(s.charAt(z)==s.charAt(j)){

if(j-z==1){

dp[z][j]=1;

}

else if(dp[z+1][j-1]==1){

dp[z][j]=1;

}

else{

dp[z][j]=0;

}

}

else{

dp[z][j]=0;

}

if(dp[z][j]==1){

count++;

}

z++;

j++;

}

}

return count;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>