2. JAVA PROGRM-BALANCED BRAKET

Write a function that accepts a string consisting entiring of brackets ({}) and returns whether it is balanced. Every "opening" bracket must be followed by a closing bracket of the same type. There can also be nested brackets, which adhere to the same rule.  
f('()[]{}(([])){[()][]}') // true  
f('())[]{}') // false

import java.util.Stack;

public class Main {

public static void main(String[] args) {

System.out.println(is\_parentheses\_balanced("()[]{}(([])){[()][]}"));

}

public static boolean matchingPeer(char open , char close){

if ( open == '(' && close == ')'){

return true;

}

if ( open == '[' && close == ']'){

return true;

}

else{

return false;

}

}

public static boolean is\_parentheses\_balanced(String equation){

char[] c = equation.toCharArray();

Stack <Character> myStack= new Stack <Character> ();

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

if(c[i]=='(' || c[i] == '[' ){

myStack.push(c[i]);

}

else if (c[i]== ')' || c[i]==']'){

if(matchingPeer(myStack.peek(),c[i]) == true){

myStack.pop();

} else {

return false;

}

}

}

if(myStack.isEmpty()){

return true;

}

else {

return false;

}

}

}

**Output:**

