You're implementing the isValid method to check if a string containing only the characters '(', ')', '{', '}', '[', and ']' is a valid parentheses sequence.

To do this correctly, you can use a **stack** to ensure each opening bracket has a matching and properly ordered closing bracket.

Here's the **correct implementation**:

import java.util.Stack;

class Solution {

public boolean isValid(String s) {

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

for (char c : s.toCharArray()) {

// Push opening brackets onto the stack

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

stack.push(c);

}

// Check closing brackets

else if (c == ')' && (stack.isEmpty() || stack.pop() != '(')) {

return false;

} else if (c == '}' && (stack.isEmpty() || stack.pop() != '{')) {

return false;

} else if (c == ']' && (stack.isEmpty() || stack.pop() != '[')) {

return false;

}

}

// Stack should be empty if all brackets are matched

return stack.isEmpty();

}

}